DISEÑO DE UNA GUÍA PARA LA ADMINISTRACIÓN DE LA SEGURIDAD DE BASES DE DATOS EN UN ENTORNO DE ORACLE 11G, APLICADA A LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC) EN LA CIUDAD DE CALI.

MARGARITA LEAL JOYA
GUSTAVO ADOLFO HERRERA ANGOLA

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD.
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
SANTIAGO DE CALI
2016

DISEÑO DE UNA GUÍA PARA LA ADMINISTRACIÓN DE LA SEGURIDAD DE BASES DE DATOS EN UN ENTORNO DE ORACLE 11G, APLICADA A LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC) EN LA CIUDAD DE CALI.

MARGARITA LEAL JOYA
GUSTAVO ADOLFO HERRERA ANGOLA

Trabajo de grado para optar al título de Especialista en seguridad Informática

Director: Salomón González García

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD.
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA ECBTI
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
SANTIAGO DE CALI
2016

CONTENIDO

pág.

TABLA DE ANEXOS

pág.

LISTA DE FIGURAS

pág.

LISTA DE TABLAS

pág.

1. INTRODUCCIÓN

La información más sensible y valorada por las empresas por lo general reposa en una base de datos, es por esto que contar con alto nivel de seguridad para éstas en un punto fundamental y a garantizar en cualquier entidad. Una base de datos con una óptima configuración de seguridad, reduce el grado de vulnerabilidad ante ataques informáticos o fallas técnicas y logra de igual manera, conseguir un nivel de respaldo mayor para responder de forma rápida ante cualquier eventualidad en la seguridad de la información.

Es por esta razón que el proceso de investigación desarrollado en el presente trabajo se enfocó inicialmente en la identificación de la mayor cantidad posible de vulnerabilidades en una la base de datos causadas por un proceso inapropiado o incompleto de configuración, para posteriormente realizar el diseño y generación de una guía de seguridad de base de datos basada en estándares y las mejores prácticas de la seguridad de la información enmarcadas en la normatividad vigente, específicamente para la versión de base de datos Oracle 11g.

La aplicación de la guía al entorno empresarial, fue realizada en la Corporación Autónoma Regional Del Valle Del Cauca, permitiendo la identificación de vulnerabilidades de la base de datos, sugiriendo la implementación de controles a partir de un proceso de forma secuencial y ordenada para cada uno de los casos identificados y a ejecutar en la configuración según las recomendaciones de la guía.

El modelo de configuración de seguridad elaborado en este proyecto, facilitará y orientará el proceso de implementación de técnicas y mecanismos de seguridad en las bases de datos, de tal forma que garanticen la disponibilidad, integridad y confidencialidad de la información; permitiendo al administrador de la base de datos enfocar sus esfuerzos en las funciones que le conciernen de una forma más eficiente y eficaz.

2. TITULO.

DISEÑO DE UNA GUÍA PARA LA ADMINISTRACIÓN DE LA SEGURIDAD DE BASES DE DATOS EN UN ENTORNO DE ORACLE 11G, APLICADA A LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC) EN LA CIUDAD DE CALI.

3. DEFINICIÓN DEL PROBLEMA

Teniendo en cuenta que las bases de datos son el medio donde se almacena y gestiona los datos de una empresa, se presenta en muchos casos que el administrador de estos medios se enfoca más en configurar la Base de Datos para su óptimo funcionamiento, buscando minimizar tiempos de respuesta en accesos y procesos transaccionales, dejando en un segundo plano la seguridad de la información. Comúnmente en la instalación y configuración de Base de datos no se cuenta con el personal idóneo en cuanto a experiencia en la administración, lo que genera posiblemente las siguientes situaciones:

- Configuración errada por un proceso de instalación estándar el cual usa más componentes de los que generalmente son pactados en las especificaciones de la licencia adquirida.
- Falta de procedimientos de encriptación para claves de acceso en la generación de copias de seguridad a nivel de servidor de almacenamiento y base de datos
- Falta de parametrización para la definición del nivel de seguridad según criticidad o volumen de información manejado por la empresa
- Deficiencia en la definición de mecanismos de control de acceso, privilegios y acceso autorizado de usuarios a nivel de base de datos.
- Falta de implementación de bitácoras y pistas de auditoria a nivel de usuarios y tablas para definición y creación de registros de auditoria.

Cuando una Base de Datos no ha tenido una adecuada configuración de seguridad y no cuenta con mecanismos de respaldo confiables ni buenas prácticas, se expone a la organización a colocar en alto riesgo su información, ya que se considera más vulnerable a ataques informáticos o fallas técnicas y no contará con mecanismos de respuesta rápidos y confiables que logren minimizar los daños causados. La pérdida de confiabilidad, integridad y disponibilidad de la información, pueden llevar a las empresas a consecuencias desastrosas a nivel financiero.

3.1. FORMULACIÓN DEL PROBLEMA

¿Cómo el diseño de la guía de administración de seguridad en la bases de datos Oracle 11g, basada en buenas prácticas y estándares de seguridad, permitirá al administrador la implementación de controles y la configuración adecuada de la bases de datos mejorar la seguridad de la informática y de información en la Corporación Autónoma Regional del Valle del Cauca (CVC) en la ciudad de Cali?.

4. JUSTIFICACIÓN

El desarrollo de esta guía para la configuración y parametrización de una base de datos Oracle 11g, basada en estándares, mecanismos de seguridad y las mejores prácticas de seguridad de la información, será una herramienta que facilite y oriente en el proceso de implementación de técnicas y mecanismos de seguridad en las bases de datos, y garanticen la disponibilidad, integridad y confidencialidad de la información de la empresa.

Esta guía será de gran utilidad para administradores de bases de datos, ingenieros, estudiantes de Ingeniería o profesionales de áreas afines que estén realizando actividades como administración de bases de datos o que estén en prácticas en temáticas relacionadas con esta actividad, permitiéndoles afinar conocimientos a partir de la documentación de las actividades propias de la administración y configuración de seguridad en las base de datos y brindando un orden en los procesos realizados.

La guía se convierte en una ayuda para el Administrador de Base de Datos, para poder implementar de una forma más eficaz niveles de seguridad y mecanismos de respaldo apropiados en las Bases de datos de una organización. Este documento permitirá al administrador de base de datos dirigir sus esfuerzos de una forma eficiente y eficaz en sus funciones y esto a su vez repercute en parte en el éxito de las empresas para mejorar su competitividad, ya que reducen el grado de vulnerabilidad ante ataques informáticos o fallas técnicas, logrando de igual forma conseguir un nivel de respaldo mayor para responder de forma rápida ante cualquier eventualidad de falla en la seguridad de la información.

5. OBJETIVOS

5.1. OBJETIVO GENERAL

Diseñar una guía de administración de seguridad en la bases de datos Oracle 11g, que permita al administrador la implementación de controles y configuración apropiada de la base de datos, basándose en buenas prácticas y estándares de seguridad que den cumplimiento a regulaciones y normativas vigentes.

5.2. OBJETIVOS ESPECÍFICOS

- Realizar un análisis de estándares y buenas prácticas aplicadas a la seguridad en Base de datos para determinar las principales características de seguridad propias de la BD Oracle, que dan cumplimiento a regulaciones y normativas de seguridad.

- Determinar la configuración y mecanismos de respaldo de una Base de datos Oracle 11g, basada en buenas prácticas y estándares de seguridad, que den cumplimiento a normativas de seguridad de: control de acceso, auditoria, protección de información por mecanismos de encriptación, mecanismos de respaldo y restauración seguros.

- Diseñar una guía de administración de seguridad en la bases de datos Oracle 11g, basándose en buenas prácticas y estándares de seguridad que den cumplimiento a regulaciones y normativas vigentes.

- Realizar pruebas que permitan identificar las vulnerabilidades de las bases de datos en los sistemas de información de la Corporación Autónoma Regional del Valle del Cauca (CVC), con el fin de proponer posibles controles que apliquen y den cumplimiento a normativas de seguridad de las bases de datos en cuanto a control de acceso, auditoria, protección de información por mecanismos de encriptación, mecanismos de respaldo y restauración seguros.

6. MARCO REFERENCIAL

6.1. ANTECEDENTES

- El proyecto denominado “AUDITORIA Y CONTROL EN ENTORNOS BAJO ORACLE 11G” presentado por David García Bastanchuri en la UNIVERSIDAD Carlos III de Madrid en la ciudad de Leganés – España 2013. En el proyecto se presentan un análisis de la auditoría sobre sistemas de información basados en el sistema gestor de bases de datos Oracle 11g. Este proyecto brinda información sobre los conceptos de auditoría informática, control interno sobre los entornos de bases de datos.

- El proyecto denominado “ESTUDIO DE CARACTERÍSTICAS SEMÁNTICAS SOBRE ORACLE 11G” presentado por Ángel Fabricio Sánchez Sarango en la UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA - Escuela de Ciencias de la Computación en la ciudad de Loja – Ecuador 2010. En el proyecto se explica cómo aplicar características semánticas que proporciona Oracle 11g. Este proyecto brinda información sobre instalación y configuración de Oracle 11g.

- El proyecto denominado “IMPLANTACIÓN DEL SISTEMA DE GESTIÓN DE SEGURIDAD DE LA INFORMACIÓN PARA LAS BASES DE DATOS ORACLE SISTEMA MUISCA DE LA SUBDIRECCIÓN DE GESTIÓN DE TECNOLOGÍA DE LA INFORMACIÓN Y TELECOMUNICACIONES EN LA DIRECCIÓN DE IMPUESTOS Y ADUANAS NACIONALES” presentado por Mario Pérez Lozano y Luis Omar Correa Visbal en la Universidad Nacional Abierta y a Distancia (UNAD) -2015. En el proyecto se presenta una propuesta de un Sistema de Gestión de Seguridad de la Información SGSI para la Dirección de Impuestos y Aduanas Nacionales DIAN enfocada especialmente a la base de datos Oracle que maneja la entidad. Este proyecto brinda información de buenas prácticas a implementar en una base de datos de una entidad, basándose en los controles seguridad de la norma ISO 27001 Anexo A.

6.2. MARCO TEÓRICO

6.2.1. Sistema de gestión de bases de datos (SGDB). Es un conjunto de programas que permiten la administración de una base de datos. Estos sistemas cuentan con mecanismos de control para administrar acceso a los datos, recuperación y respaldo en caso de fallos y opciones de monitoreo, entre otras características, que dependen tanto de del hardware como del tipo software que se configura para tal fin. Un SGDB cuenta con las características para ser

configurado de tal forma que suministre garantías de seguridad en la información en una organización[1].

Entre los sistemas de Gestión de Base de Datos comerciales se encuentra Oracle Database, que es un SGDB de tipo Objeto relacional desarrollado por Oracle Corporation. Oracle se destaca por proporcionar soluciones robustas para la protección de la información almacenada en la base de datos, facilitando el cumplimiento de normativas de seguridad de la información.

6.2.2. Normas de seguridad de la información. Entre los estándares y normas más sobresalientes a nivel mundial orientados a la seguridad de la información están los diseñados por la Organización Internacional de Normalización ISO (International Organization for Standardization) y la Comisión Electrotécnica Internacional IEC (International Electrotechnical Commission), se cuenta con la Norma ISO/IEC 27000, esta norma brinda un marco referencial y las definiciones que aportan la base para la implantación de un Sistema de Gestión de Seguridad de la Información; Dentro de las normas que están en la familia ISO/IEC 27000, se destacan:

ISO 27001: la norma ISO 27001[2] brinda una guía para la implementación de un sistema de gestión de seguridad de la información; Basado en la gestión y análisis de riesgos para su tratamiento y mitigación. Al implementar la norma ISO27001 para la creación del Sistema de Gestión de la Seguridad de la Información (SGSI), en la empresa, se estará asegurando que se lleven a acabo 4 principales actividades relacionadas al SGSI:

- Establecer el sistema.
- Implementar y operar el sistema.
- Mantener y mejorar el sistema.
- Monitorear y revisar el sistema.

Siendo la gestión del riesgo el enfoque central de la norma ISO 27001, obliga a la organización, a implementar el estándar, definiendo y realizado un inventario de los activos informáticos, un análisis de riesgo por activo, para determinar el tratamiento a seguir en pro de mitigar el riesgo, identificando los controles apropiados a implantar para reducir los riesgos.

Esto con la finalidad de asegura la continuidad y conservación de la seguridad de la información, atrás de la implantación, uso y actualización constante de los controles adoptados en la metodología PDCA: Planificar, Hacer, Verificar, Actuar.

---

[1] (SARRIA, Francisco Alonso.Sistemas de Información Geográfica. Capítulo 9 Sistemas de Gestión de Bases de datos y SIG. Universidad de Murcia)

[2] (ISO 27000.es, El portal de ISO 27001 en Español [En línea]. 2016, Disponible en: http://www.iso27000.es/iso27000.html)

ISO 27002: Es una guía para la validación e implementación de mejorares a la seguridad de la información de una entidad; Que presenta objetivos de seguridad a perseguir junto con los controles relacionados, se muestran como una guía de mejores prácticas[3]. En la última versión de la norma se describe los siguientes dominios de control:

05. Política de Seguridad
06. Organización de la Seguridad de Información
07. Gestión de Activos
08. Seguridad ligada a los Recursos Humanos
09. Seguridad Física y del Entorno
10. Gestión de Comunicaciones y Operaciones
11. Control de Accesos
12. Adquisición, Desarrollo y Mantenimiento de Sistemas de Información
13. Gestión de Incidentes de Seguridad de la Información
14. Gestión de Continuidad del Negocio
15. Conformidad

6.2.3. Seguridad informática y seguridad de la información. La seguridad informática es el área que se encarga de plantear normas, técnicas y procedimientos para proteger la información, a nivel de recursos de software, hardware y también humanos, en los que se apoyan los procesos críticos de una organización. La seguridad de la información está relacionada con el asegurar la confidencialidad, integridad y disponibilidad en la manipulación de la información, buscando minimizar riesgos y amenazas, con la aplicación de buenas prácticas, estándares y normativas de seguridad[4]. La norma internacional ISO 27000[5], brinda definiciones de conceptos de seguridad de la información como son:

- **Disponibilidad:** Asegurar que los usuarios autorizados siempre tengan acceso a la información que requieran. Se garantice que la información sea puntual y con sus respectivos privilegios para acceder a la información.
- **Integridad:** garantizar que la información del sistema no sea alterada por usuarios no autorizados con el fin de evitar la pérdida de consistencia de información.
- **Confidencialidad:** busca que la información privada no se revele a usuarios o terceros no autorizados.

[3] (ISO 27002.es, El portal de ISO 27002 en Español [En línea]. 2016, Dispoible en: http://www.iso27000.es/iso27002.html)
[4] (MIFSUD, Elvira. Introducción a la seguridad informática - Seguridad de la información / Seguridad informática. disponible en: http://recursostic.educacion.es/observatorio/web/ca/software/software-general/1040-introduccion-a-la-seguridad-informatica?start=1)
[5] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario.[En línea]. 2016, Dispoible en: http://www.iso27000.es/glosario.html)

La seguridad de la información se apoya en la seguridad informática para cumplir con su objetivo de garantizar la Confidencialidad, Integridad y Disponibilidad de la información. A su vez la seguridad informática se apoya en los lineamientos planteados por la seguridad de la información, para proyectar los mecanismos de seguridad en la infraestructura tecnológica y de comunicaciones a implementar en la organización.

6.2.4. Normas/ Estándares en seguridad en base de datos. Existen diferentes estándares de seguridad de la información a nivel mundial que son aplicables a las base de datos. Entre las más destacadas se encuentran:

- **COBIT:** Objetivos de control para la información y tecnologías relacionadas. Es un conjunto de mejores prácticas y controles que permite auditar la gestión de los sistemas de información y tecnología desarrollado por la Asociación de Auditoría y Control del Sistema de Información ISACA. [6]

- **BIS:** Instituto Británico de Estándares. Brinda normas que tiene como objeto la estandarización de procesos, enfocado en la característica de seguridad de la información: Auditoria, certificación y formación. [7]

- **Normas ISO:** International Standards Organization. Proporciona normas para la administración, mantenimiento y continuidad de la seguridad de la información; enmarcada en las normas de la familia ISO/IEC 27000, como la ISO/IEC 27001 y su anexo ISO/IEC 27002.[8]

- **HIPAA:** Health Insurance Portability and Accountability Act (Ley de Portabilidad y Contabilidad de los Seguros de Salud), conjunto de reglas relacionada con el manejo de la confidencialidad de la información. [9]

6.2.5. Seguridad en base de datos Oracle. La Base de datos Oracle dispone de opciones y soluciones, que facilitan el cumplimiento a regulaciones y normativas de seguridad, según la documentación oficial de Oracle[10], en su documentación técnica describe mecanismos de seguridad a nivel de:

- Sistema de Control de Accesos, Gestión de privilegios y Revisión de derechos de acceso a usuarios. Oracle permite el manejo de opciones de administración de

[6] (ISACA, COBIT 5 Spanish. [En línea]. 2016, Dispoible en: http://www.isaca.org/spanish/Pages/default.aspx)
[7] (BIS, Sobre BSI. [En línea]. 2016, Dispoible en: http://www.bsigroup.com/es-ES/Sobre-BSI/)
[8] (ISO 27000.es, El portal de ISO 27001 en Español [En línea]. 2016, Disponible en: http://www.iso27000.es/iso27000.html)
[9] (HHS, Office for Civil Rights. Health Information Privacy. [En línea]. 2016, Disponible en: http://www.hhs.gov/hipaa/index.html)
[10] (ORACLE CORPORATION., Documento técnico de Oracle, Marzo de 2011 -Seguridad y cumplimiento rentables de Oracle Database 11g versión 2)

usuarios, privilegios y roles. Para esto adicionalmente cuenta con el producto de seguridad Oracle 'Database Vault'.

- Para protección contra ataques de SQL Injection, Oracle dispone del productos como 'Bind Variables', que permite el manejo de datos dinámicos dentro de instrucciones de consulta SQL; Oracle cuenta con opciones para validar las direcciones IP desde donde se puede acceder a la información. Soluciones como Oracle Virtual Private Database (VPD) y Oracle Label Security (OLS), disminuyen la cantidad de información que puede ser afectaba por SQL Injection.

- Gestión de Comunicaciones y Operaciones, Registro de auditorías, Supervisión de uso del sistema, Protección de la información de registro. Para esto Oracle cuenta con mecanismos de activación de pistas de auditorías. Cuenta con una serie de comandos para auditar los cambios en la base de datos, tanto DDL (create, alter, drop) como DML (insert, update, delete).

- Controles Criptográficos. Oracle dispone de controles a este nivel como: Transparent Data Encryption (TDE) que permite la encriptación transparente de datos. Network encryption: Para el cifrado de red, con estándares de encriptación (RC4, DES, AES). Integrity of information: Asegurando que los mensajes no se modifiquen en su tránsito. Strong authentication, soportando diferentes metos de autenticación como Kerberos, RADIUS (Remote Authentication Dial-In User Service), Secure Sockets Layer (with digital certificates), PKI .

- Salvaguarda de los registros de la organización. Oracle cuenta con diferentes opciones para realizar respaldo y restauración de base de datos. Cuenta con un modo de configuración de la base de datos en ARCHIVELOG, que permite aumentar la potencia de herramientas como RMAN (Recovery Manager) para realizar copias y restauración de una base de datos mucho más flexible y con diferentes opciones. Oracle cuenta con productos que permiten asegurar los Backuop en cinta, como Oracle Secure Backup, o encriptar archivos EXPORT, usando métodos como Oracle Transparent Data Encryption o Passphrase de Oracle Data Pump.

6.2.6. Vulnerabilidades de las bases de datos. Dentro de las principales vulnerabilidades que se encuentran en las Bases de Datos están:

- Mecanismos de seguridad débiles en la configuración de perfiles. Perfiles con demasiados privilegios y que no son utilizados.

- Autenticación como usuario administrador SYSDBA, sin asignación de contraseña. Una vez instalada la base de datos Oracle, se crea por defecto el usuario que creó la Base de datos rol SYSDBA. Este usuario puede acceder la base de datos sin necesidad de usar contraseña.

- Usuarios y contraseñas creados por defecto en la instalación de Oracle, algunos con privilegios de DBA, y que facilitan el acceso de atacantes cuando no son cambiados o eliminados.

- Algoritmos de verificación de contraseña débiles, que permiten al atacante la fácil identificación de credenciales usuarios, permitiendo acceder e penetrar en la seguridad de la Base de datos.

- Debilidades en los aplicativos que faciliten ataques de SQL Injection[11], como cadenas de conexión a la base de datos con usuario y contraseña explicitas en el código fuente , sentencias SQL construidas dinámicamente , mensajes de error que revelen información de la Base de datos. Otro aspecto que facilita los ataques SQL Injection, es el bajo nivel de seguridad en los procedimientos realizados por el lenguaje de programación propio de ORACLE, PL/SQL.

- Desbordamiento de buffer, causadas por fuentes de entrada masivas con valores diferentes o muy superiores a los que se espera en la aplicación. Se debe tener en cuenta la actualización de parches en el software de la base de datos que brinda el proveedor, para solucionar vulnerabilidades encontradas en sus versiones.

- Utilización de mecanismos débiles de cifrado de la información de la base de datos.

- Débil configuración de privilegios a usuarios sobre objetos de la base de datos o sistema.

- Proceso TNS listener (transparent network substrate) desprotegido, ya que es posible acceder de forma fácil a información de la instancia de la Base de datos (bases de datos almacenadas, IP de servidores de Base de datos, puertos).

- Falta de capacitación en las políticas de seguridad de la empresa a los usuarios de la base de datos en la administración de contraseñas.

- Falta de mecanismos de monitoreo en la base de datos débiles, que permitan identificar a tiempo posibles fallas y aplicar las correcciones respectivas.
- Débil configuración de mecanismos de auditoria, que no faciliten el rastreo de ataques y detectar amenazas de seguridad, esto coloca en alto riesgo la seguridad de los datos.

[11] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 5)

- Cuando los datos críticos del negocio en la base datos no cuentan con controles y mecanismos de seguridad de cifrado dejándola expuesta a amenazas.

- Configuraciones de respaldo débiles, que no permitan ejecutar la recuperación rápida y con la menor pérdida posible de información en caso de falla o ataque a la información.

6.2.7. Tipos de ataques a las bases de datos. Los ataques a las bases de datos suelen ser originados, por vulnerabilidades de las aplicaciones web, entre los tipos de ataques más frecuentes son:

- Ataques por técnicas de inyección de scripts
- Ataques por SQLinjection
- Ataques de Path Transversal
- Ataques por técnicas de inyección de código
- Ataques de inyección de ficheros

Ataques por técnicas de inyección de scripts: Este tipo de ataque se presenta al inyectar código o datos en una línea de comando o consulta; Entre estos tipos de ataques se encuentran: Cross site scripting (XSS), Cross site request forgery (CSRF) y clickjacking. [12]

- *Cross site scripting (XSS):* consiste en introducir código Javascript a una aplicación WEB con vulnerabilidades, con la finalidad de robar información, apropiarse de sesiones activas, corromper el navegador web y como consecuencia afectar la integridad del sistema.

  Un ejemplo de este tipo de ataques es la suplantación de identidad o phishing, a través del direccionamiento del acceso a páginas, sitios web, servidores, este tipo de ataque se base en la confianza del usuario tiene sobre el sitio frecuentado.

- *Cross site request forgery (CSRF):* falsificación de petición en sitios cruzados. En este tipo de ataque CSRF, usa un usuario o dirección IP de usuarios validados para el sistema, generando que el usuario realice acciones no deseadas en sitios remotos. Este tipo de ataque, dado que se realiza con un usuario propio, dependiendo de los privilegios con lo que cuente el usuario en el sistema seria la magnitud del ataque que puede causar.

[12] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 5)

- *Clickjacking*: este tipo de ataque pretende engañar al usuario para que accedan a través de un clic sobre links. De esta forma el atacante obtiene información, logrando causar ataques de tipo CSRF (Cross-site request forgery).Este tipo ataque superpone páginas, en lugares donde habitualmente el usuario accede.

Ataques de inyección SQL: Esta técnica se basa en ejecutar operaciones directamente a la base de datos[13]. Este tipo de técnica, aprovecha debilidades en:

- Comprobación de parámetros de entrada
- Comprobación de parámetros utilizados en códigos SQL
- Construcción de sentencias SQL, dinámicas
- En la construcción de código PL/SQL en el caso de Oracle

Con esta técnica se posibilita al atacante:

- Acceder a información sin autorización de la a base datos tales como registros y objetos.
- Elevación de privilegios, accediendo con credenciales de usuarios con mayores privilegios y alterando permisos.
- Denegación de servicio, la modificación de sentencias SQL, puede llevar a cabo acciones que provoquen destrucción a nivel de: borrado de datos y/ó objetos, detener servicios. Al igual que ejecutar sentencias que generen lentitud en las respuestas del sistema hasta colapsarlo.
- suplantación de usuarios: cuando el atacante puede acceder a la información de credenciales de usuarios, es posible que pueda tomar alguna y ejecutar procesos usando las credenciales robadas.

Ataques de Path Transversal: Los ataques que se basan en esta técnica, van dirigidos a lograr conseguir acceso a ficheros del servidor, carpetas fuera de donde se encuentra alojada la aplicación, explota la vulnerabilidad que se ocasiona cunado no existe seguridad en cuanto a la validación de usuarios en la aplicación, o no cuenta con controles de manejo de errores controlados, que impidan la salida de errores por defecto, que normalmente muestran rutas del archivo afectado donde se provocó el error[14]. De esta forma se accede a ficheros a los cuales un usuario no debería tener acceso, logrando acceder a información crítica.

[13] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 5)
[14] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 54)

Ataques de inyección de código: Los ataques de inyección SQL consisten en la modificación de datos de ingreso a través de inserción de consultas o sentencia SQL. [15]

Con este tipo de ataque se pretende por lo general obtener información la base de datos, modificar datos, realizar operaciones de administración, entre las acciones más buscadas se destacan

- Saltar restricciones de acceso.
- Elevar privilegios.
- Obtención de información de la base de datos
- Detener servicios de del gestor de base de datos.
- Ejecución de sentencias SQL dentro del servidor.

Ataques de inyección de ficheros: Este tipo de ataques permite la inclusión de archivos remotos o locales, debido a debilidades en el código de programación de la aplicación, por falta de filtros. Esto permite que se puedan modificar parámetros o archivos del sistema, comprometiendo la seguridad. Por ejemplo modificación de archivo de contraseñas.[16]

Este tipo de ataques puede darse por inclusión de archivo remotos, inclusión de archivos locales o Webtrojans. Estos últimos que se dan cuando la página permite subir archivos como imágenes, documentos, pdf, videos y no cuentan con mecanismo de comprobación del tipo de archivo enviado, permitiendo la llegada al servidor de archivos malintencionados.

6.2.8. Pruebas de seguridad en base de datos. Para escanear vulnerabilidades en base de datos, es posible realizarlas a través de pruebas de caja negra o pruebas de caja blanca.

Pruebas caja negra: Son pruebas donde se simulan técnicas de ataques donde es posible hallar vulnerabilidades en la seguridad de la información que pueden comprometer datos críticos o las operaciones normales del sistema[17]. Se basan en pruebas de entrada y salida de datos, no es necesario contar con accesos autorizados ni conocer el funcionamiento del objetivo al que se le va a realizar la prueba de ataque.

---

[15] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 26)
[16] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 2 Ataques a aplicaciones web. Universidad Abierta de Cataluña. p. 56)
[17] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 4 Auditoría y desarrollo seguro. Universidad Abierta de Cataluña. p. 6)

Ventajas:

- Proporciona un acercamiento más real a las posibles amenazas a los que puede estar expuesto el sistema.
- Se obtienen hallazgos por medio de información externa o pública.
- Para llevar a cabo este tipo de pruebas no es necesario contar con usuarios y claves autorizadas para su acceso, no es necesario conocer el funcionamiento del sistema.

Desventajas:

- La recopilación de los datos iniciales para ejecutar la prueba de ataque, en la mayoría de los casos es de un alto costo.
- Se requiere de una experticia del auditor o atacante en las diferentes técnicas a utilizar para las pruebas y encontrar vulnerabilidades

Principales técnicas de ataque:

- Cross-site scripting
- Spoofing
- Inyección de SQL
- Inyección de código
- Basadas en validación de entrada / salida
- Desbordamiento de buffer
- Basadas en secuestro de sesiones
- Basadas en sniffing
- Escalamiento de privilegios

Pruebas caja blanca: En este tipo de pruebas se valida que los controles y políticas de seguridad que se encuentran implementadas estén acorde a las normativas vigentes y a las exigencias del negocio[18]. En estas pruebas se cuenta principalmente con permisos para acceder normalmente a los sistemas, base de datos, código fuente del entorno que se disponga para la prueba.

Ventajas:

- Es posible realizar pruebas en un nivel detallado y de esta misma forma encontrar fallas concretas de configuración.
- Es posible realizar recomendaciones precisas para los hallazgos encontrados.

[18] (CEBRIÁN,José María Alonso, et al. Seguridad en bases de datos, Módulo 4 Auditoría y desarrollo seguro. Universidad Abierta de Cataluña. p.10)

Desventajas:

- Se debe tener conocimiento de la funcionalidad del objetivo a realizar la prueba
- Solo es posible realizarlo contando con las credenciales de acceso a los sistemas y base de datos.
- Debido a que son pruebas específicas y detalladas no se tiene un panorama de las posibles fallas por accesos externos no autorizados.

Auditoria tipo test de penetración: Este tipo de auditoria es posible aplicarla para evaluar la seguridad de bases de datos a partir de test de penetración. [19]

Estrategias de prueba de penetración:

- **Orientadas a objetivo:** Estas pruebas se realizan en compañía del grupo de tecnologías informáticas o afines, y se enfoca en la detección de un objetivo específico. Las pruebas son realizadas de manera que toda persona que tenga relación con el objetivo identificado, pueda aportar o validar sobre la forma o uso de la técnica aplicada.

- **Comprobación externa:** Este tipo de pruebas se enfoca en detectar el nivel de acceso que se puede obtener ingresando a los dispositivos externos, como son servidores de nombres de dominio (DNS), de correo, web o cortafuegos.

- **Comprobación interna:** Este tipo de pruebas se realizan con usuarios que cuenten con los permisos de un empleado de la entidad y desde una terminal interna, con el fin de validar el cortafuego. Con este tipo de pruebas se validan las posibles consecuencias de la realización de ataques desde el interior de la entidad.

- **Pruebas a ciegas:** Este tipo de pruebas se realiza con una mínima cantidad de información, lo que genera una gran inversión de tiempo, esfuerzo y recursos.

6.2.9. Software para realizar pruebas a base de datos. Para la realización de pruebas específicas de auditoria, cuando existen aplicaciones web en la organización que interactúan con la Base de datos, se encuentran herramientas comerciales y opensource que ayudan a detectar vulnerabilidades que se explotan a través de plugin. Entre estas herramientas se encuentran:

[19] (EcuRed, Prueba de penetración. [En línea]. 2016, Dispoible en: https://www.ecured.cu/Prueba_de_penetraci%C3%B3n)

- *W3af*: Herramienta de auditoría web OpenSource, multiplataforma, que permite encontrar vulnerabilidades web y explotarlas. Está basado en plugins elaborados en lenguaje Python[20]. Entre las vulnerabilidades que detectan se encuentran:

    - Manipulación de navegador por usuario externo (CSRF) Cross site request forgery.
    - Suplantación de dominios a través de XSS.
    - Inyecciones de código la cual permite modificar o extraer información desde un almacén de datos (SQL Injection, XPath Injection y LDAP Injection).
    - Desbordamiento de buffer por sobrecarga (Buffer overflows)
    - Ejecución de ficheros externos al servidor simulando una ubicación local (Remote file inclusion).

- *SQLMAP*: Herramienta OpenSource, basada en líneas de comandos que automatiza los procesos para detectar y explotar vulnerabilidades de SQL Injection y extracción de información en Base de datos[21]. soporta base de datos en MySQL, Oracle, PostgreSQL, Microsoft SQL Server.

- *Acunetix:* es una herramienta que permite escanear vulnerabilidades en aplicaciones web, sobre plataformas Windows[22]. Entre las vulnerabilidades que detecta y explota se encuentran:

    - Inyecciones de código el cual permite modificar o extraer información desde un almacén de datos (CRLF injection)
    - Manipulación de navegador por usuario externo a través de código (CSS) Cross-Site Scripting
    - Captura e ingreso a directorios con claves débiles (Directory Traversal)
    - Ejecución de comandos o inyección de código (Code execution).

- *NESSUS*: Es una herramienta que permite la identificación de vulnerabilidades de un servidor de base de datos sin importar el sistema operativo en el cual se encuentre instalada. Cuenta con una estructura cliente servidor, que facilita el uso de hosting o ips vinculadas o accedidas desde el entorno de red[23]. Esta Herramienta dispone de una base de

[20] (W3af, SQL Injection, Cross-Site Scripting and much more. [En línea]. 2016, Disponible en: http://w3af.org/)
[21] (SQLMAP, Introduction [En línea]. 2016, Disponible en: http://sqlmap.org/)
[22] (ACUNETIX, Scan your websites [En línea]. 2016, Disponible en: https://www.acunetix.com/)
[23] (NESSUS, Tenable network security [En línea]. 2016, Disponible en: https://www.tenable.com/products/nessus-vulnerability-scanner)

plugins, que tiene como referencia para llevar a cabo el escaneo de vulnerabilidades. Posee una interfaz que arroja resultados de forma gráfica y con convenciones de colores para identificar el nivel de severidad de las vulnerabilidades encontradas, a su vez que las clasifica por familia de plugin a la que hace referencia. Es posible realizar configuraciones para el nivel de detalle del escaneo a realizar así como de las políticas utilizadas.

Las herramientas Software que pueden ser usadas para la ejecución de las pruebas a la seguridad, en cuanto a la configuración y administración de la base de datos y ejecución de sentencias y generación de informes de estado de seguridad en plataforma Oracle están:

- Toad DBA Suite for Oracle: Esta herramienta comercial que cuenta con una versión libre bajo restricciones, permite realizar tareas de pruebas a las bases de datos, cuenta con una interfaz gráfica con múltiples opciones que facilitan las actividades de la administración de Base de datos[24], características:

    - Permite el monitoreo de secciones por usuario y aplicativo de forma gráfica.
    - Permite la generación de reportes de rendimiento, configuración, listado de esquemas de la base de datos y vulnerabilidades de seguridad a nivel de bases de datos.
    - Brinda la posibilidad de realizar diagnósticos predictivos de la base de datos característica que permite la identificar posibles problemas por rendimiento.
    - Permite la realización diagnóstico de rendimiento de Oracle RAC (Oracle Real Application Clusters) a través de niveles de instancias, clústeres e interconexiones.
    - Se basa en el lenguaje SQL (Structured Query Language), cuenta con un editor de consultas.

- Oracle SQL Developer: Es una herramienta de Oracle, gratuita, que permite el desarrollo y gestión de base de datos Oracle[25], características:

    - Permite la creación de código PL/SQL y ejecución de consultas SQL.
    - Dispone de una consola para la realización de tareas de DBA, que permite la generación de copias de seguridad, mediante el uso de RMAN (Oracle Recovery Manager) es posible realizan scripts para restaurar y recuperar una base de datos.
    - Cuenta con una interfaz de informes.

[24] (QUEST, Join Toad World. [En línea]. 2016, Disponible en: https://www.quest.com/toad/)
[25] (ORACLE, SQL Developer. [En línea]. 2016, Disponible en: http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index-097090.html)

- Dispone de opciones para migración de base de datos Oracle

- Oracle Enterprise Manager 11g: Es una conjunto de herramienta propias de Oracle para el monitorio bases de datos a varios niveles a través de tipos de conexiones: Standalone (modo autónomo), y OMS (Oracle Management Server) [26], las herramientas con que disponen brindan la opción de:

    - Administración de procesos de negocios
    - Dispone de opciones de consola para la administración y realización de cambios de configuración entre las aplicaciones de Oracle usadas.
    - Implementación de metodología para la identificación rápida de cuellos de botella.
    - Administración grupos de aplicaciones las cuales permiten la integración entre aplicaciones y gestión
    - Permite la administración de las configuraciones para la automatizar los procesos propios de área de tecnología e Información.

6.3. MARCO CONCEPTUAL

Para el desarrollo de proyecto se medirán variables que están directamente involucradas con vulnerabilidades, amenazas y riesgos de la seguridad de la información, en cuanto a su *Disponibilidad, integridad y Confidencialidad* específicamente en el tratamiento en las Bases de datos. A continuación se describe algunos conceptos en el marco de la seguridad de la información según la Organización Internacional de estándares ISO, dentro del estándar ISO 27000[27]:

- **Amenaza:** "Causa potencial de un incidente no deseado, que puede provocar daños a un sistema o a la organización[28]".
- **Riesgo**: "Posibilidad de que una amenaza concreta pueda explotar una vulnerabilidad para causar una pérdida o daño en un activo de información. Suele considerarse como una combinación de la probabilidad de un evento y sus consecuencias[29]."

[26] (ORACLE, Oracle Enterprise Manager 11g. [En línea]. 2016, Disponible en: http://www.oracle.com/technetwork/es/oem/grid-control/overview/index.html)
[27] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario.[En línea]. 2016, Dispoible en: http://www.iso27000.es/glosario.html)
[28] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario. Amenaza.[En línea]. 2016, Dispoible en: http://www.iso27000.es/glosario.html)
[29] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario. Riesgo. [En línea]. 2016, Dispoible en: http://www.iso27000.es/glosario.html)

- **Vulnerabilidad**: “Debilidad de un activo o control que puede ser explotada por una o más amenazas[30].”
- **Confidencialidad**: “Debilidad de un activo o control que puede ser explotada por una o más amenazas[31].”
- **Disponibilidad**: “Propiedad de la información de estar accesible y utilizable cuando lo requiera una entidad autorizada[32].”
- **Integridad**: “Propiedad de la información relativa a su exactitud y completitud[33].”

6.4. MARCO LEGAL

El Congreso de la República de Colombia en el 2009 promulgó la ley 1273, donde se clasificaron los delitos informáticos en una lista de actuaciones relacionadas con el manejo de la seguridad de la información y los sistemas informáticos. La ley 1273[34] promulga: “Por medio del cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado –denominado “De la Protección de la información y de los datos”- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones”.

A continuación se muestran artículos relevantes de la ley, principalmente del Capítulo I publicado en la página oficial de la Alcaldía de Bogotá: [35]

“CAPITULO. I De los atentados contra la confidencialidad, la integridad y la disponibilidad de los datos y de los sistemas informáticos:

- Artículo 269A: ACCESO ABUSIVO A UN SISTEMA INFORMÁTICO. El que, sin autorización o por fuera de lo acordado, acceda en todo o en parte a un sistema informático protegido o no con una medida de seguridad, o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

---

[30] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario. Vulnerabilidad.[En línea]. 2016, Dispoible en: http://www.iso27000.es/glosario.html)
[31] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario.Confidencialidad.[En línea]. 2016, Disponible en: http://www.iso27000.es/glosario.html)
[32] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario. Disponibilidad.[En línea]. 2016, Disponible en: http://www.iso27000.es/glosario.html)
[33] (ISO 27000.es, El portal de ISO 27001 en Español. Glosario. Integridad.[En línea]. 2016, Disponible en: http://www.iso27000.es/glosario.html)
[34] (Alcaldía Mayor de Bogotá D.C., LEY 1273 DE 2009. [En línea]. 2016, Disponible en: http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=34492)
[35] (Alcaldía Mayor de Bogotá D.C., LEY 1273 DE 2009. [En línea]. 2016, Disponible en: http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=34492)

- Artículo 269B: OBSTACULIZACIÓN ILEGÍTIMA DE SISTEMA INFORMÁTICO O RED DE TELECOMUNICACIÓN. El que, sin estar facultado para ello, impida u obstaculice el funcionamiento o el acceso normal a un sistema informático, a los datos informáticos allí contenidos, o a una red de telecomunicaciones, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con una pena mayor.

- Artículo 269C: INTERCEPTACIÓN DE DATOS INFORMÁTICOS. El que, sin orden judicial previa intercepte datos informáticos en su origen, destino o en el interior de un sistema informático, o las emisiones electromagnéticas provenientes de un sistema informático que los trasporte incurrirá en pena de prisión de treinta y seis (36) a setenta y dos (72) meses.

- Artículo 269D: DAÑO INFORMÁTICO. El que, sin estar facultado para ello, destruya, dañe, borre, deteriore, altere o suprima datos informáticos, o un sistema de tratamiento de información o sus partes o componentes lógicos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

- Artículo 269E: USO DE SOFTWARE MALICIOSO. El que, sin estar facultado para ello, produzca, trafique, adquiera, distribuya, venda, envíe, introduzca o extraiga del territorio nacional software malicioso u otros programas de computación de efectos dañinos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

- Artículo 269F: VIOLACIÓN DE DATOS PERSONALES. El que, sin estar facultado para ello, con provecho propio o de un tercero, obtenga, compile, sustraiga, ofrezca, venda, intercambie, envíe, compre, intercepte, divulgue, modifique o emplee códigos personales, datos personales contenidos en ficheros, archivos, bases de datos o medios semejantes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.”

7. MARCO METODOLÓGICO

7.1. METODOLOGÍA DE LA INVESTIGACIÓN

El enfoque que tiene la investigación es cuantitativo, ya que pretende hacer la medición de la vulnerabilidad, amenazas y riesgos enmarcado en las características de confidencialidad, integridad y disponibilidad de la información, como resultado de la aplicación de recomendaciones para la configuración y administración de una base de datos.

El tipo de investigación tiene componente teórico, ya que inicialmente se requiere realizar un estudio de las vulnerabilidades, amenazas y riesgos con las que cuentan actualmente las Bases de Datos en entornos Oracle 11g. Para posteriormente aplicar pruebas a la base de datos física en la cual se va a desarrollar el proyecto y de esta forma validar el estado de vulnerabilidad de la base de datos. Por último se elabora el manual para la configuración óptima de seguridad de la Base datos Oracle 11g.

La propuesta de investigación es aplicada, ya que busca resolver problemas que se generan en la práctica de la administración y configuración de una base de datos, concretamente en entornos Oracle 11g.

La investigación es de tipo explicativa, dado que intenta exponer la relación entre las vulnerabilidades existentes y los ataques que pueden presentarse en una base de datos a partir de la configuración de seguridad que se le haya efectuado.

Es descriptiva porque sobre ella se realizan una serie de mediciones de las variables relacionadas con la seguridad de la información, para poder identificar cómo se va a llevar a cabo la configuración de las bases de datos, aplicando mecanismos y estándares de seguridad de la información. Se requiere de un conocimiento previo de conceptos de seguridad de la información enfocada a base de datos en entornos Oracle 11g.

7.1.1. Universo y muestra. Todos los usuarios que administran, acceden e interactúan con la Base de Datos almacenada en los servidores ubicados en la sede central de la Corporación Autónoma Regional del Valle del Cauca, se identifican como la población.

La muestra será, los usuarios encargados de la administración de la Base de datos y los usuarios creados para el uso de los aplicativos.

7.1.2. Instrumentos de recolección de información. Se utilizaron listas de chequeo, para validar el estado inicial y final de la configuración de la Base de

datos en entorno Oracle 11g, basados en estándares y normativas de seguridad de la información.

8. DESARROLLO DEL PROYECTO

8.1. BUENAS PRÁCTICAS DE SEGURIDAD EN BASE DE DATOS ORACLE 11G

De acuerdo al primer objetivo planteado para el desarrollo del proyecto, se identifican las buenas prácticas de seguridad informática aplicada a Bases de Datos y su aplicabilidad en Oracle 11g, relacionando el cumplimiento a regulaciones y nomas de seguridad a las que dan respuesta. Entre las regulaciones a las que dan cumplimiento el uso de buenas prácticas en Oracle están: Sarbanes-Oxley, HIPAA, PCI, European Directive 2002/58/EC. A continuación se presenta un cuadro detallando las buenas prácticas en seguridad de Base de datos con Oracle 11g y citando las regulaciones a las que da cumplimiento:

Tabla 1. Mejores Prácticas Oracle y su cumplimiento con regulaciones de seguridad

Mejores prácticas	**Oracle**	**Regulación**
Control de acceso		
División de tareas de administración	Cuenta con herramientas para la división de tareas de administración como Oracle Database Vault. Al igual que con características propias embebidas en la base de datos.	SOX, PCI, HIPAA, European Directive 2002/58/EC
No permitir el acceso como sysdba sin contraseña, usuario administrador del SO y BD	Permite configurar en la base de datos. perfiles de usuario de sistema operativo y restringir permisos elevados en usuarios.	SOX, PCI, HIPAA, European Directive 2002/58/EC
Fuente: Seguridad y cumplimiento rentables de Oracle. Database 11g versión 2. Documento técnico de Oracle. Disponible en: http://www.oracle.com/technetwork/es/database/enterprise-edition/documentation/seguridad-y-cumplimiento-11gr2-2247594-esa.pdf		

Tabla 2. (Continuación)

Mejores prácticas	Oracle	Regulación
Control de acceso		
Cambiar o deshabilitar el acceso a usuarios y contraseñas creados por defecto durante la instalación.	Permite configurar en la base de datos para deshabilitar usuarios. Oracle cuenta con vistas del diccionario de datos que permiten conocer los usuarios creados por defecto, configuración de perfiles, roles y permisos asignados para su monitoreo y administración.	SOX, PCI, HIPAA, European Directive 2002/58/EC
No dejar usuarios y claves escritos en aplicaciones o procedimientos de la BD	Oracle cuenta con mecanismos para realizar tareas de validación y enmascaramiento de objetos de la BD	SOX, PCI, HIPAA, European Directive 2002/58/EC
Protección de datos		
Utilización de mecanismos de cifrado y gestión de seguridad para contraseñas	Oracle permite configurar parámetros el profile asignado en la creación de usuarios, administrando el comportamiento de las contraseñas, como el tiempo de validez; cantidad de intentos fallidos de acceso a la base de datos antes de bloquearse; tiempo para solicitar cambio de contraseña; bloqueo de direcciones IP según el número de intentos de conexión a la base de datos fallida; asignación de funciones para que se lleve a cabo la verificación de contraseñas seguras validando longitud, uso de mayúsculas y minúsculas, alfanuméricas, no permitir que sea igual al nombre de usuario, y demás características de complejidad que se requieran. La base de datos Oracle cuenta con una opción para crear y habilitar una función que trae consigo una serie de reglas para aplicar en la creación de contraseñas.	SOX, PCI, HIPAA, European Directive 2002/58/EC

Tabla 3. (Continuación)

Mejores prácticas	Oracle	Regulación
Protección de datos		
Utilización de mecanismos de cifrado para backup	Oracle cuenta con la opción Oracle Secure Backup que permite gestionar cintas seguras y protección de datos en general. Opciones de Oracle Advanced Security, permite la encriptación para Backup creados con RMAN	SOX, PCI, HIPAA, European Directive 2002/58/EC
Utilización de mecanismos de cifrado para objetos de la base de datos, procedimientos, funciones, paquetes	Oracle cuenta con opciones para encriptar objetos de la base de datos que contengan código PL/SQL, por ejemplo haciendo uso de la herramienta wrap, que permite ejecutar sentencias para encriptar este tipo de objetos de la base de datos	SOX, PCI, HIPAA, European Directive 2002/58/EC
cifrado a nivel de columnas de tablas, para el caso de aplicaciones que lo requieran por cumplimiento de regulaciones	Permite el cifrado de columnas y datos confidenciales de tablas, con la opción Oracle TDE (Transparent Data Encryption).	SOX, PCI, HIPAA, European Directive 2002/58/EC
Auditoria y monitoreo		
Ejecución y realización de auditoria para operaciones del usuario sys	Oracle cuenta con mecanismos para llevar a cabo auditoría de base de datos a través de vistas propias de Oracle.	SOX, PCI, HIPAA
Realización de auditoria fina para tablas sensibles	Oracle cuenta con la herramienta Oracle Audit Vault para la realización de auditoria. De igual forma características propias de la base de datos permite configurar opciones para habilitar este tipo de auditoria.	SOX, PCI, HIPAA

Tabla 4. (Continuación)

Mejores prácticas	Oracle	Regulación
Auditoria y monitoreo		
Ejecución y realización de auditoria estándar	Oracle cuenta con la herramienta Enterprise Manager para llevar a cabo seguimiento de actividades de auditoria de forma gráfica. Así como consulta a las tablas o vistas de auditoria del sistema que es posible recuperar y monitorear con la ejecución de sentencias sql.	SOX, PCI, HIPAA
Realizar la auditoria de las sentencias DDL	Oracle cuenta con la herramienta Configuration Scanning la cual permite la realización de sondeo de trazado sobre las actividades realizadas en la base de datos. De igual forma la ejecución de consultas directas a las tablas de auditoria del sistema o configuración de triggers en las tablas criticas permitiría realizar controles de auditoria, adicionando funcionalidades como envío de correos al administrador del a base de datos.	SOX, PCI, HIPAA
Realizar de rutinas periódicas de escaneos de actividades realizadas a nivel de usuarios y sentencias ejecutadas o comandos empleados	Oracle cuenta con la herramienta Configuration Scanning la cual permite la realización de sondeo de trazado sobre las actividades realizadas en la base de datos. Al igual que la herramienta Enterprise Manager para llevar a cabo seguimiento de actividades de auditoria de forma gráfica	SOX, PCI, HIPAA
Mecanismos de respaldo y restauración		
Realización de verificación del estados de las copias generadas	Oracle cuenta con herramientas como Oracle Recovery Manager (RMAN) y Oracle Secure Backup, para la gestión y realización de copias de seguridad de la información. Recovery Manager (RMAN) Y Oracle Data Pump Export/Import	SOX, HIPAA

Tabla 5. (Continuación)

Mejores prácticas	Oracle	Regulación
Mecanismos de respaldo y restauración		
Gestión interna de soportes y recuperación; Realización de copias de respaldo de manera periódica y programada; Realización de montaje de copias de respaldo en entornos de prueba, para garantizar la integridad de la información y confiabilidad del procedimiento de carga de Backups	Oracle cuenta con herramientas como Oracle Recovery Manager (RMAN), Oracle Data Pump Export/Import y Oracle Secure Backup, para la gestion y realización de copias de seguridad de la información. Recovery Manager (RMAN) Y Oracle Data Pump Export/Import.	SOX, HIPAA

8.2. CONFIGURACIÓN DE BUENAS PRÁCTICAS EN UNA BASE DE DATOS ORACLE 11G

8.2.1. Control de acceso

- **Consulta de parámetros de configuración de la base de datos:** Según la documentación oficial de Oracle[36], se puede visualizar los parámetros de configuración asignados a la base de datos, a partir de la consulta “SELECT * FROM V$SYSTEM_PARAMETER;”, Se puede visualizar los parámetros de configuración asignados a la base de datos.

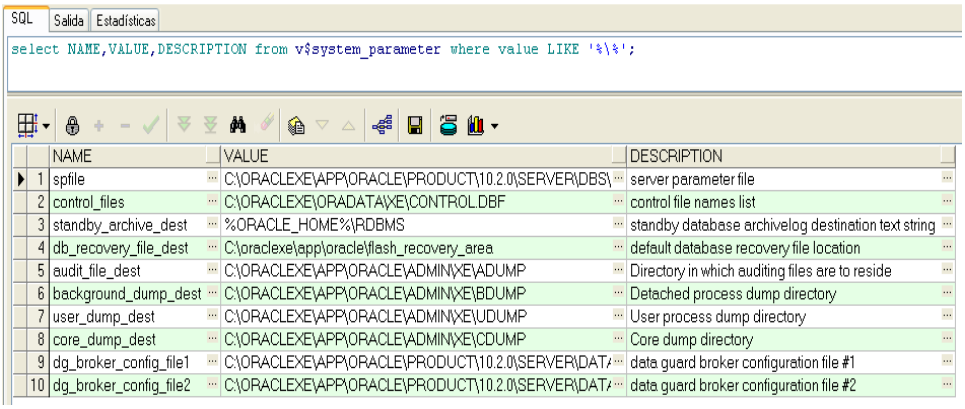
[36] (ORACLE CORPORATION. Oracle Help Center, Database Reference: V$SYSTEM_PARAMETER. [En línea]. 2016 Disponible en: https://docs.oracle.com/cd/E11882_01/server.112/e40402/dynviews_3097.htm#REFRN30275)

Para identificar las rutas de almacenamiento de los archivos propios de Oracle se usa la siguiente consulta sql:

*select * from v$system_parameter where value LIKE '%\%';*

Estas ubicaciones deben estar protegidas principalmente a nivel de sistema operativo y/o red, para evitar la degradación o fallos a nivel de base de datos como consecuencia de manipulación de archivos o directorios.

Figura 1 Consulta de configuración de la base de datos



	NAME	VALUE	DESCRIPTION
1	spfile	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DBS\	server parameter file
2	control_files	C:\ORACLEXE\ORADATA\XE\CONTROL.DBF	control file names list
3	standby_archive_dest	%ORACLE_HOME%\RDBMS	standby database archivelog destination text string
4	db_recovery_file_dest	C:\oraclexe\app\oracle\flash_recovery_area	default database recovery file location
5	audit_file_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\ADUMP	Directory in which auditing files are to reside
6	background_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\BDUMP	Detached process dump directory
7	user_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\UDUMP	User process dump directory
8	core_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\CDUMP	Core dump directory
9	dg_broker_config_file1	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DAT/	data guard broker configuration file #1
10	dg_broker_config_file2	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DAT/	data guard broker configuration file #2

Fuente: Propia

- **Revisión de usuarios por defecto.** Durante la instalación de una base de datos Oracle, se crean usuarios por defecto, con privilegios y contraseñas que son de fácil acceso, debido a que se encuentran en la literatura de instalaciones de Oracle. Para evitar que la base de datos sea accedida por personas no autorizadas o ajenas a la empresa, se requiere llevar a cabo tareas para cambiar contraseñas, eliminar privilegios o bloquear los usuarios que son creados por defecto en la instalación.

  Para identificar los usuarios que fueron creados en el proceso de instalación de la base de datos y que aun cuentan con la contraseña que se asigna por defecto, se consulta la vista '*DBA_USERS_WITH_DEFPWD*'. Según la documentación oficial de Oracle[37] para identificar de forma rápida estos usuarios se ejecuta la siguiente consulta sql:

  *"SELECT * FROM DBA_USERS_WITH_DEFPWD;"*

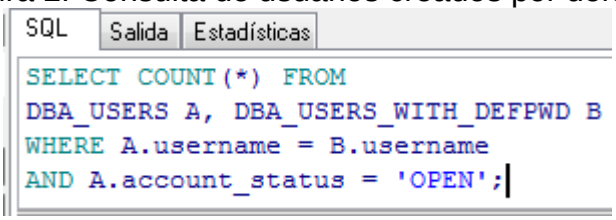
[37] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS_WITH_DEFPWD. [En línea]. 2016 Disponible en:https://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5074.htm)

De los usuarios que se encuentren, ejecutando la anterior consulta es necesario conocer, cuántos de ellos se encuentran aún activos en la base de datos, y que puedan ser utilizados por una persona no autorizada. Para esto es necesario hacer uso de la vista DBA_USERS, según la documentación oficial de Oracle[38], esta vista cuenta con la información de las cuentas de usuarios de la base de datos y el estado de la cuentas.

Haciendo uso de la vista *DBA_USERS y la tabla DBA_USERS_WITH_DEFPWD , es posible* conocer de los usuarios creados por defecto, cuantos se encuentran activos en la base de datos, La sentencia sql a ejecutar es:

SELECT COUNT(*) FROM DBA_USERS A, DBA_USERS_WITH_DEFPWD B WHERE A.username = B.username AND A.account_status = 'OPEN';

Figura 2. Consulta de usuarios creados por defecto



Fuente: Propia

Los usuarios que se encuentren activos con contraseñas por defecto, es necesario, cambiar la contraseña o bloquearlos según se requiera. Para bloquearlos la sentencia sql a ejecutar es:

ALTER USER [NombreUsuario] ACCOUNT LOCK;

Para llevar a cabo un cambio de contraseña, la sentencia sql a ejecutar es:

alter user [Nombre_usuario] identified by [Nueva_contraseña];

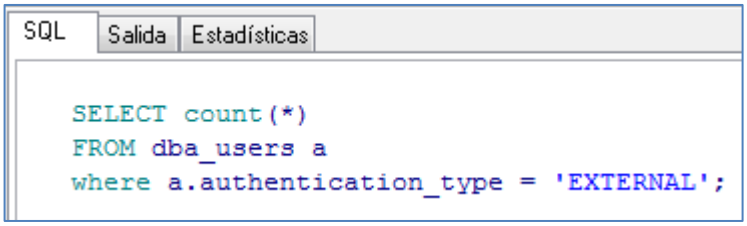
- **Privilegios de usuario de sistema operativo con permisos elevados en la base de datos.** Los usuarios con autenticación de sistema operativo, según la documentación oficial de Oracle[39], se pueden identificar por ser creados normalmente con la característica de tipo de autenticación “EXTERNAL”, que les permite acceder a la base de datos sin ingresar contraseña.

[38] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS [En línea]. 2016, Dispoible en: https://docs.oracle.com/cd/E18283_01/server.112/e17110/statviews_5081.htm)
[39] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS [En línea]. 2016, Dispoible en: https://docs.oracle.com/cd/E18283_01/server.112/e17110/statviews_5081.htm)

Por defecto en la instalación se crea el usuario de autenticación de sistema operativo con privilegios de administrador. Debido a esto el tipo de usuario no requiere autenticación para acceder a la base de datos, se debe validar sus privilegios en la base de datos, ya que normalmente el administrador de la base de datos no es el mismo administrador del sistema operativo, esto para garantizar un nivel mayor de seguridad, es recomendable que el usuario del sistema operativo solo cuente con los privilegios de conexión. Para validar usuarios con tipo de autenticación se ejecuta la siguiente sentencia sql:

"SELECT count(*) FROM dba_user A WHERE a.authentication_type = 'EXTERNAL'".

Figura 3. Consulta de usuarios autenticación EXTERNAL



Fuente: Propia

El usuario de sistema operativo son creados con el prefijo 'OPS$', lo cual también es posible consultarlo dentro de la misma vista de DBA_USER, con la siguiente consulta sql: *SELECT * FROM dba_users a where a.username like '%OPS$%'.*

Según la documentación oficial de Oracle,[40] para validar si el usuario de sistema operativo cuenta con privilegios de administrador, es posible consultarlo por la vista dba_role_privs , ejecutando la siguiente sentencia sql:

```
select count(*)
from dba_users a, dba_role_privs b
where a.username = b.GRANTEE
and a.username like '%OPS$%'
and b.GRANTED_ROLE = 'DBA';
```

Si se encontrara que el usuario de sistema operativo cuenta con privilegios de administrador, es necesario retirarlos ya que es un usuario con autenticación external, lo que implica que accede a la base de datos sin necesidad de escribir la contraseña, lo que implicaría un riesgo que un atacante solo conociendo el

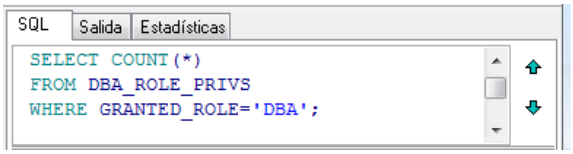
[40] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_ROLE_PRIVS. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_4206.htm)

nombre del usuario de sistema operativo ingrese a la base de datos con privilegios de administrador sin que se le solicite contraseña.

- **Validación de usuarios con permisos de administrador**. Para conocer los usuarios con rol de administrador, según la documentación oficial Oracle[41] , se hace una consulta sobre la vista de Oracle DBA_ROLE_PRIVS, que muestra los privilegios asignados a usuarios y roles de la base de datos. La consulta a la vista se hace referenciando el rol DBA, para conocer los usuarios con permisos elevados. La sentencia sql a ejecutar:

```
SELECT COUNT(*) FROM DBA_ROLE_PRIVS WHERE GRANTED_ROLE='DBA';
```

Figura 4. Consulta de usuarios con permiso DBA



Fuente: Propia

Según la documentación oficial de Oracle[42], para quitar un privilegio a un usuario se ejecuta la sentencia:

REVOKE role FROM {user, | role, |PUBLIC}

Es recomendable realizar la ejecución de esta sentencia periódicamente ya que permite buscar inconsistencias en el número de usuarios con privilegios DBA, que posiblemente hayan sido creados por un atacante.

Realizar tareas de validación de privilegios de los usuarios periódicamente, permite controlar el acceso no autorizado a la base de datos y tomar las acciones pertinentes para garantizar que los permisos de los usuarios sobre los objetos sean los mínimos requeridos.

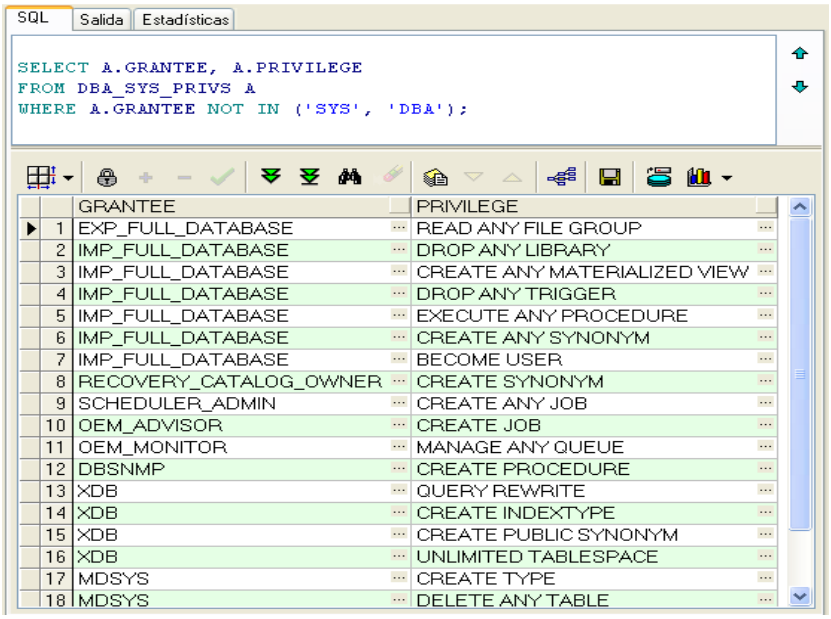
Otro objeto de la base de datos que permiten consultar información relacionada a los roles del sistema, según la documentación oficial de Oracle[43], es la vista

---

[41] Ibíd.
[42] (ORACLE CORPORATION. Oracle Help Center, Database Reference: REVOKE. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_9020.htm)
[43] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_SYS_PRIVS. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5036.htm)

DBA_SYS_PRIVS . En la siguiente figura se muestra un ejemplo de una consulta a esta vista:

Figura 5. Consulta de usuarios con permiso DBA por la vista DBA_SYS_PRIVS



	GRANTEE	PRIVILEGE
1	EXP_FULL_DATABASE	READ ANY FILE GROUP
2	IMP_FULL_DATABASE	DROP ANY LIBRARY
3	IMP_FULL_DATABASE	CREATE ANY MATERIALIZED VIEW
4	IMP_FULL_DATABASE	DROP ANY TRIGGER
5	IMP_FULL_DATABASE	EXECUTE ANY PROCEDURE
6	IMP_FULL_DATABASE	CREATE ANY SYNONYM
7	IMP_FULL_DATABASE	BECOME USER
8	RECOVERY_CATALOG_OWNER	CREATE SYNONYM
9	SCHEDULER_ADMIN	CREATE ANY JOB
10	OEM_ADVISOR	CREATE JOB
11	OEM_MONITOR	MANAGE ANY QUEUE
12	DBSNMP	CREATE PROCEDURE
13	XDB	QUERY REWRITE
14	XDB	CREATE INDEXTYPE
15	XDB	CREATE PUBLIC SYNONYM
16	XDB	UNLIMITED TABLESPACE
17	MDSYS	CREATE TYPE
18	MDSYS	DELETE ANY TABLE

Fuente: Propia

- **Validación de usuarios con privilegios para ejecutar comandos ddl.** Según la documentación oficial de Oracle44, la base de datos cuenta con la vista SYSTEM_PRIVILEGE_MAP, en la que se puede consultar todos los privilegios con los que cuenta Oracle, los cuales pueden ser asignados a usuarios o roles.

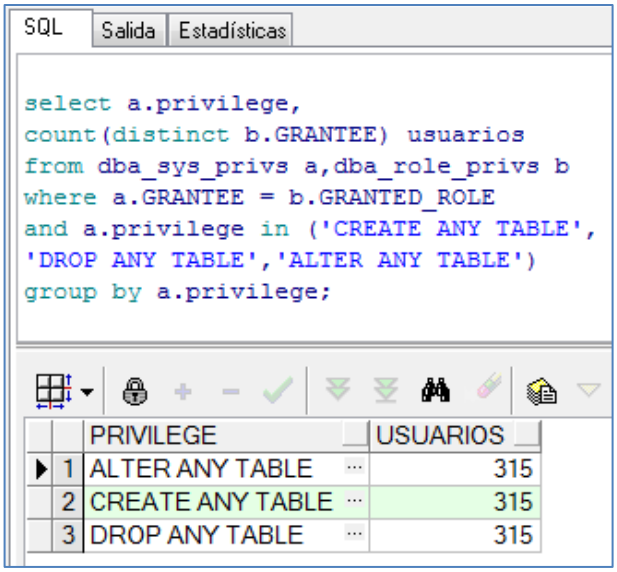
  En la vista SYSTEM_PRIVILEGE_MAP se encuentran los privilegios para ejecutar comandos DDL (data definition language), como como CREATE, ALTER y DROP. Según la documentación oficial de Oracle[45], este tipo de comandos permite definir estructuras de objetos de la base de datos.

  Para validar los usuarios que actualmente cuentan con permisos en el sistema para ejecutar comandos DDL como CREATE, ALTER y DROP sobre todas las tablas, se ejecuta la siguiente sentencia sql:

[44] (ORACLE CORPORATION. Oracle Help Center, Database Reference: SYSTEM_PRIVILEGE_MAP. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5158.htm)
[45] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: Types of SQL Statements. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_1001.htm)

```
select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by a.privilege;
```

Figura 6. Usuarios con permisos DDL

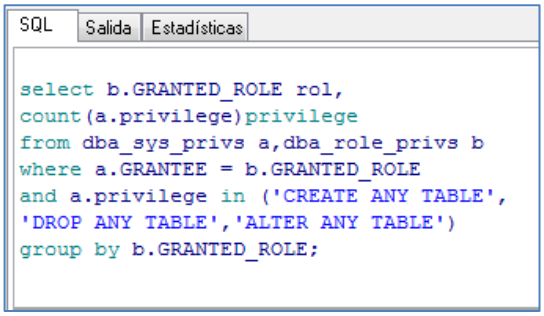


Fuente: Propia

Para consultar los roles que cuentan con permisos para crear, alterar o borrar tablas, se ejecuta la siguiente sentencia SQL:

```
select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by b.GRANTED_ROLE;
```

Figura 7. Roles con permisos DDL



Fuente: Propia

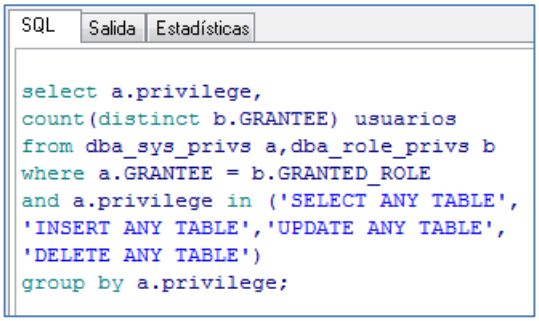
Con el resultado de estas consultas, el administrador de la base de datos, puede identificar y monitorear los usuarios que deben tener este tipo de privilegios.

- **Validación de usuarios con privilegios para ejecutar comandos dml.** Las sentencias de lenguaje de manipulación de datos (Data Manipulation Language DML), según la documentación oficial de Oracle[46], son utilizadas para realizar tareas de consulta, inserción, modificación y eliminación de los datos contenidos en las bases de datos, por esta razón se recomienda conocer los usuarios que cuentan con los permisos para realizar este tipo de actividades sobre objetos del sistema y la ejecución de monitoreo que permita la identificación de los usuarios existentes y validación de los mismos para conocer la cantidad de usuarios con permisos en el sistema para ejecutar comandos DML (Data manipulation language), como SELECT, INSERT, UPDATE y DELETE sobre todas las tablas, se ejecuta la siguiente sentencia sql:

select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by a.privilege;

[46] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: Types of SQL Statements. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_1001.htm)

Figura 8. Usuarios con permisos DML

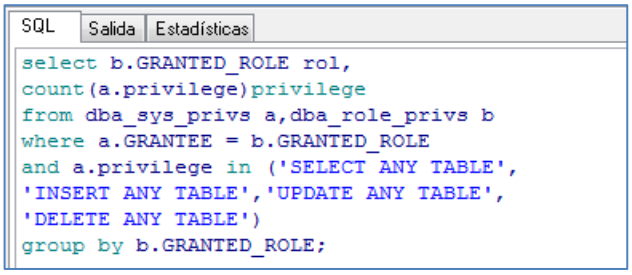


Fuente: Propia

Para consultar los roles que cuentan con permisos para seleccionar, modificar, actualizar o borrar tablas, se ejecuta la siguiente sentencia SQL:

select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by b.GRANTED_ROLE;

Figura 9. Roles con permisos DML



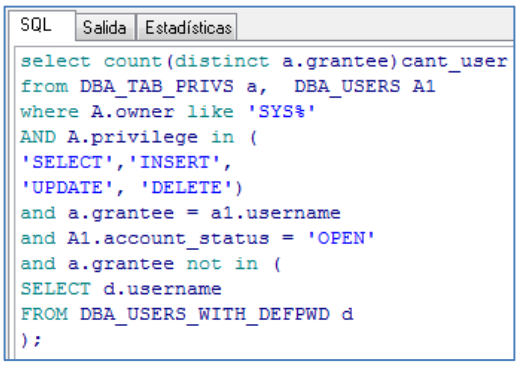
Fuente: Propia

Otra forma de conocer los usuarios con permisos para ejecutar comandos DML sobre objetos del sistema es haciendo referencia la vista DBA_TAB_PRIVS, según la documentación oficial de Oracle[47] esta vista describe todos los privilegios asignados a los objetos en la base de datos referenciando al propietario del objeto. Se ejecuta la siguiente sentencia SQL:

[47] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_TAB_PRIVS. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5046.htm)

```
select count(distinct a.grantee)cant_user
from DBA_TAB_PRIVS a,  DBA_USERS A1
where A.owner like 'SYS%'
AND A.privilege in (
'SELECT','INSERT',
'UPDATE', 'DELETE')
and a.grantee = a1.username
and A1.account_status = 'OPEN'
and a.grantee not in (
SELECT d.username
FROM DBA_USERS_WITH_DEFPWD d
);
```

Figura 10. Consulta a usuarios en la vista DBA_TAB_PRIVS



Fuente: Propia

Con el resultado de estas consultas, el administrador de la base de datos, puede identificar y monitorear los usuarios que deben tener este tipo de privilegios y tomar la acción de revocarlos o bloquearlos en el caso que sea necesario.

- **Políticas de contraseñas.** Según la documentación oficial de Oracle[48], la consulta de perfiles de la base de datos y su configuración es posible conocerla por medio de la vista DBA_PROFILES, que se basa en la tabla sys.profile$. Cada perfil de la base de datos define los límites de los recursos de la base de datos que se la asignaran a los usuarios que se creen, a nivel de contraseñas (password_parameters) y de límites de recursos asignados de servidor (resource_parameters). Según la documentación oficial de Oracle[49], Los principales parámetros de PASSWORD,en los perfiles son los siguientes:

[48] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: DBA_PROFILES. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_4175.htm)

[49] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm)

- PASSWORD_LIFE_TIME: especifica el tiempo de vigencia de la contraseña.
- PASSWORD_GRACE_TIME: especifica el tiempo límite en días para la modificación de la contraseña, antes de que se bloquee la cuenta.
- PASSWORD_REUSE_TIME: tiempo en días para utilizar el mismo password
- PASSWORD_REUSE_MAX: Permite definir el número máximo de veces que es posible reusar la misma clave.
- FAILED_LOGIN_ATTEMPTS: número máximo de intentos fallidos de ingreso de la clave, antes de que se bloquee la cuenta de usuario, se recomienda tres intentos como máximo
- PASSWORD_LOCK_TIME : Especifica el número de días en la que una cuenta estará bloqueada después de cumplir el número de intentos fallidos consecutivos para acceder. Si no se especifica un valor, por defecto es un día.
- PASSWORD_VERIFY_FUNCTION: En este parámetro permite asignar una función para establecer la complejidad de la contraseña que crean los usuarios. Por ejemplo exigir una longitud mínima, que no se asigne el mismo nombre del usuario, que cuente con caracteres especiales, entre otras características que se puedan asignar a la función.

Cuando este parámetro PASSWORD_VERIFY_FUNCTION es null, no se cuenta con ninguna regla para la creación de contraseñas. Según la documentación oficial de Oracle[50], la base de datos cuenta por defecto con una función, 'verify_function_11G', para asignar en este parámetro, que es posible crearla ejecutando, con usuario sys, el archivo utlpwdmg.sql que se encontraría en: $ORACLE_HOME/rdbms/admin.

Para cambiar el valor del parámetro PASSWORD_VERIFY_FUNCTION, de tal forma que tome una función de verificación de contraseñas, según la documentación oficial de Oracle[51] se utiliza *ALTER PROFILE,* ejecutando la siguiente sentencia sql:

ALTER PROFILE default LIMIT
PASSWORD_VERIFY_FUNCTION verify_function_11G;

[50] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Configuring Authentication. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/network.111/b28531/authentication.htm)
[51] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: ALTER PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_2007.htm)

De igual forma es posible asignar una función personalizada, construyendo su código en PL/SQL y posteriormente alterar el perfil en su parámetro *PASSWORD_VERIFY_FUNCTION, asignando la nueva función*

Un ejemplo de configuración de perfiles de password, con una función personalizada (*Validar_Pass*) para validación de contraseñas seria:

CREATE OR REPLACE PROFILE Politica_Pass LIMIT
PASSWORD_LIFE_TIME 45
PASSWORD_GRACE_TIME 7
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 1
PASSWORD_VERIFY_FUNCTION Validar_Pass;

Adicionalmente se pueden establecer controles a nivel se servidor configurando los parámetros de resource_parameters, según la documentación oficial de Oracle[52], estos parámetros están más relacionados con la configuración de optimización de la base de datos. En la siguiente tabla se detallan algunos de estos parámetros:

Tabla 6. Descripción de parámetros resource_parameters

CAMPO	**DESCRIPCION**
SESSIONS_PER_USER	Numero límite de sesiones simultáneas de un usuario
CPU_PER_SESSION	Especifica el límite de tiempo de CPU para una sesión, expresado en centésimas de segundos.
Fuente: ORACLE CORPORATION. Oracle Help Center,Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm	

[52] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm)

Tabla 7. (Continuación)

CAMPO	DESCRIPCIÓN
CPU_PER_CALL	Especifica el límite de tiempo de CPU para una llamada a la base de datos (un análisis sintáctico, ejecutar, o ir a buscar), expresado en centésimas de segundos.
CONNECT_TIME	Especifica el límite de tiempo total transcurrido para una sesión
IDLE_TIME	Especifica los períodos permitidos de tiempo de inactividad continua durante una sesión, expresado en minutos.
LOGICAL_READS_PER_SESSION	Especifica el número permitido de bloques de datos en una sesión de lectura, incluyendo los bloques leídos de la memoria y el disco.
LOGICAL_READS_PER_CALL	Especifica el número permitido de bloques de datos leídos por una llamada para procesar una instrucción SQL (un análisis sintáctico, ejecutar, o ir a buscar).
PRIVATE_SGA	Especifica la cantidad de espacio privado que una sesión puede asignar en el shared pool del SGA.
COMPOSITE_LIMIT	Especifica el costo total de recursos para una sesión, expresado en unidades de servicio. Calculando las unidades de servicio como una suma ponderada de los parámetros: CPU_PER_SESSION, CONNECT_TIME, LOGICAL_READS_PER_SESSION, y PRIVATE_SGA

- **Acceso restringido a información de bd remotas.** Para permitir que usuarios accedan a objetos de una base de datos remota con privilegios restringidos, sin estar creado como usuario en esa base de datos, es aconsejable el uso de bdlink. Según la documentación oficial de Oracle[53] de esta forma se accede a una base remota con un usuario que no está creado en esa base de datos, limitando sus permisos. Para la creación del DBLlink, se

[53] (ORACLE CORPORATION. Oracle Help Center, Database Administrator's Guide: Database Links. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28310/ds_concepts002.htm#i1007709)

debe tener en cuenta que se ingresen datos de conexión de un usuario existente en la base de datos a la que se quiere crear la conexión remota.

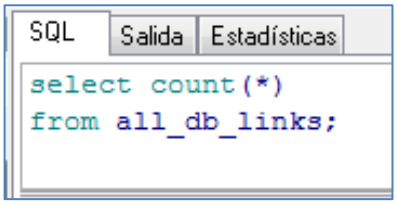
CREATE SHARED PUBLIC DATABASE LINK [NOMBRE_DB_LINK]
CONNECT TO [NOMBRE_DEL_USUARIO_DE_BASE_DATOS] IDENTIFIED BY VALUES ':1'
AUTHENTICATED BY [CLAVE_DEL_USUARIO] IDENTIFIED BY VALUES ':2'
USING '//[IP_DE_BASE_DATOS]:[PUERTO]/[NOMBRE_BASE_DATOS]';

Teniendo en cuenta que se convierte en una buena práctica para permitir acceso a una base de datos remota sin crear usuario local en esa base de datos, es aconsejable que el administrador de la base de datos, realice tareas periódicas para validar si se cuenta con esa buena práctica y a su vez validar que no se hayan creado DBLlink que no hayan sido autorizadas. Para consultar los DBLlink existentes en la base de datos se ejecuta la siguiente sentencia sql:

SELECT * FROM DBA_DB_LINKS;

Para conocer la cantidad de DBLlink, creados, se ejecuta: *select count(*) from DBA_DB_LINKS;*

Figura 11. Cantidad de DBLINK creados en la base de datos



Fuente: Propia

Buenas prácticas en el desarrollo de las aplicaciones:

- ✓ Validar que dentro de código fuente de los programas no haya usuarios y contraseñas explicitas.
- ✓ Evitar incluir código PL dentro del código fuente, procurar hacer uso de llamados a packages de la BD y estos a su vez deben pasar por procesos de encriptación.
- ✓ Evitar incluir en código fuente nombres de directorios , tablas, ID, de la BD
- ✓ Dentro de las políticas de seguridad de la organización , se debe incluir tareas periódicas que respondan a controles preventivos a la seguridad de la Base de datos como:

- La ejecución de tareas de pententing, para validar vulnerabilidades de las aplicaciones. Haciendo uso de herramientas como Sqlmap para pruebas de pentest a aplicaciones en ambiente WEB
- La validación periódica de privilegios de usuario, teniendo en cuenta privilegios de administrador, al igual que se debe contar con procedimientos precisos de la revocación de permisos de usuarios retirados.

8.2.2. Protección de datos. La confidencialidad de la información, toma una mayor relevancia en los procesos donde involucra el manejo de datos sensibles dentro de los sistemas informáticos, dada la vulnerabilidad que tiene al estar expuestos en la red. Oracle cuenta con mecanismos de seguridad a nivel de encriptación de los objetos de la base de datos. A continuación uno de estos mecanismos.

- **A nivel de código pl/sql almacenada en la base de datos**. Dentro de las buenas prácticas de programación en pl/sql se recomienda empaquetar los objetos que son usados para ejecutar un proceso en común, o porque el conjunto e interacción de estos objetos permite ejecutar alguna funcionalidad del negocio. De esta forma se facilita la administración y monitoreo del código PL/SQL utilizado.

El código PL/SQL, que se almacena en las bases de datos, es recomendable ocultarlo, principalmente para los objetos que manejan código sensible o crítico del negocio. De esta forma se protege el código fuente de la competencia del negocio, se evita daños voluntarios o involuntarios del código, y se protege a su vez propiedad intelectual del propio programador.

Oracle cuenta con herramientas que permiten ocultar información correspondiente a código PL/SQL almacenado en la BD, ya sea funciones, procedimientos o paquetes. Según la documentación oficial de Oracle[54], cuenta con la utilidad WRAP, es una de las funcionalidades con las que cuenta Oracle para llevar a cabo esta tarea, por medio de mecanismos de encriptación.

Según la empresa líder de consultoría en soporte y formación en Oracle, Burleson Colsulting[55], los pasos a seguir para encriptar objetos de la base de datos que contienen código PL/SQL son:

[54] (ORACLE CORPORATION. Oracle Help Center, Database PL/SQL Language Reference: A Wrapping PL/SQL Source Code. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/appdev.111/b28370/wrap.htm)
[55] (Burleson Colsulting. Oracle Wrap Utility. [En línea]. 2016, Disponible en:http://www.dba-oracle.com/t_wrap_utility.htm)

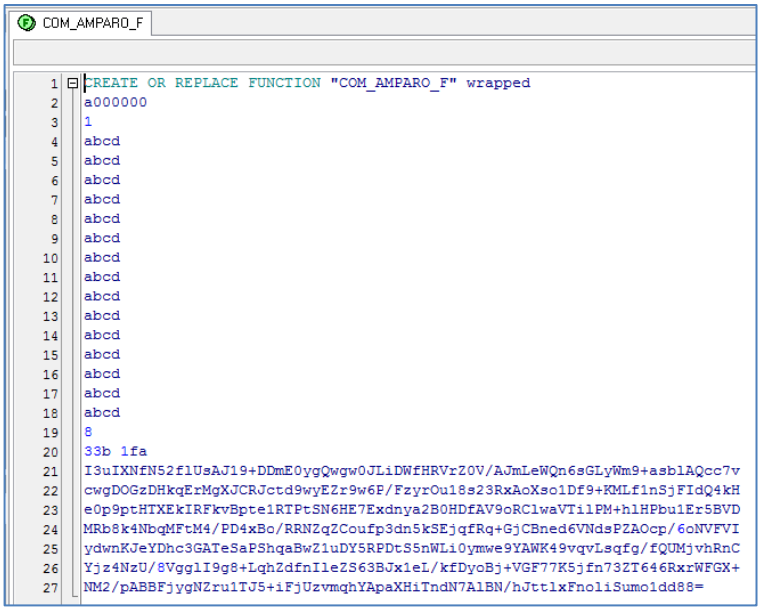
- Se identifica el objeto que almacena el código PL/SQL en la base de datos a encriptar, generando el archivo SQL para su creación en la base de datos. En el caso de los paquetes, se recomienda ocultar solamente el cuerpo, ya que la cabecera es preferible dejarla legible, para identificar la descripción de la funcionalidad del objeto, facilitando cualquier procedimiento de pruebas o calidad en la lógica de programación.
- Haciendo uso del ejecutable wrap, por consola, se ejecuta la siguiente instrucción:

  WRAP iname=NombreArchivoObjetoPL.sql
  Lo que genera un archivo de texto con extensión .plb. De esta forma se crea el archivo NombreArchivoObjetoPL.plb que contiene el código fuente convertido.
- El código creado en el archivo *.plb, se compila en una consola de SQL*plus para crear el objeto encriptado en la BD.

  SQL> @ NombreArchivoObjetoPL.plb

  Al buscar nuevamente el objeto en la base de datos se visualiza de la siguiente forma:

Figura 12. Ejemplo de objeto de base de datos encriptado



Fuente: Propia

Es recomendable como buena práctica de programación en PL/SQL, que se maneje versionamiento del código creado, utilizando repositorios de código, ya

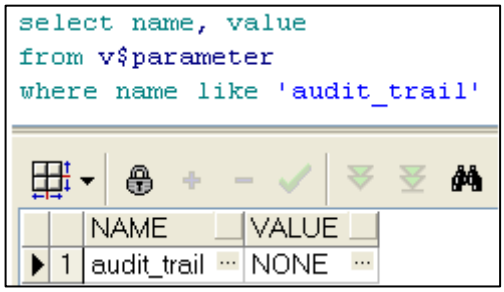
que las modificaciones al mismo se deben hacer sobre el código original repitiendo el mismo proceso para compilar.

8.2.3. Auditoria y monitoreo. Para garantizar la confiabilidad de la información debemos contar con técnicas o métodos que permitan realizar la trazabilidad de la misma, con el fin de identificar cambios, para esto es necesario la configuración de mecanismos de auditoria en una base de datos, ya que permite realizar seguimientos al uso de la base de datos, posibilitando llevar a cabo acciones preventivas o correctivas al identificar actividades en la base de datos que puedan estar produciendo algún riesgo. Entre las acciones que se registran en la auditoria de la base de datos, cuando está activa están: auditoria de inicios de sesión, auditoria de acceso a objetos y auditoria de acciones sobre objetos de la base de datos.

- **Activación de auditoria de Oracle.** Según la documentación oficial de Oracle[56], la información de las auditorias en Oracle es almacenada en diccionario de Base de Datos dentro de la tabla SYS.AUD$. Para que los datos de auditoria sean guardados, es necesario que se encuentre habilitada en el parámetro de configuración *'audit_trail'*. Para validar si la instancia de la Base de datos, tiene activa la auditoria, se consulta el parámetro de la base de datos :'audit_trail', ejecutando la siguiente instrucción SQL:

```
select name, value
from v$parameter
where name like 'audit_trail'
```

Figura 13. Visualizar el estado de la auditoria propia de Oracle



Fuente: Propia

[56] (ORACLE CORPORATION. Oracle Help Center, Database Reference: AUDIT_TRAIL. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28320/initparams017.htm)

Los valores que puede tomar el parámetro *'audit_trail'* son[57]:

- ✓ none: deshabilita la auditoría de la base de datos.
- ✓ db: habilita la auditoría, almacenando los datos en taba SYS.AUD$.
- ✓ os: habilita la auditoría de la base de datos, donde el Sistema Operativo se encarga de la auditoria de los sucesos auditados
- ✓ bd, extended: habilita la auditoría, almacenando los datos en SYS.AUD$, y adicionalmente se registran datos en la columna SQLBIND y SQLTEXT de la tabla SYS.AUD$.
- ✓ xml: habilita la auditoría, las actividades registradas se escriben en archivos XML del Sistema Operativo
- ✓ xml, extended habilita la auditoría, las actividades registradas se escriben en formato XML del sistema operativo, se incluyen valores de

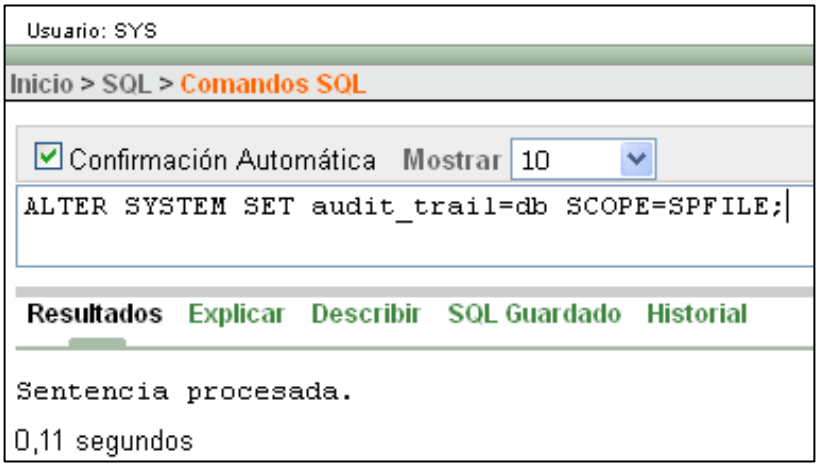
Por defecto en la instalación, Oracle trae por defecto el valor 'none' en el parámetro *'audit_trail'.*

Los pasos para habilitar la auditoria de la base de datos, según la documentación oficial de Oracle, son los siguientes: [58]

- Se ejecuta La instrucción sql:

ALTER SYSTEM SET audit_trail=db SCOPE=SPFILE;

Figura 14. Habilitar auditoria Oracle

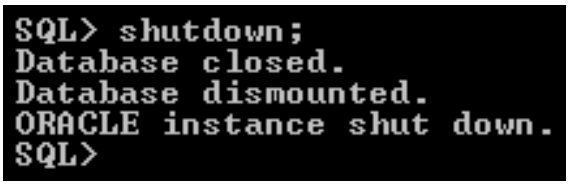


Fuente: Propia

[57] (ORACLE CORPORATION. Oracle Help Center, Database Reference: AUDIT_TRAIL. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28320/initparams017.htm)
[58] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Verifying Security Access with Auditing. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/network.112/e36292/auditing.htm#DBSEG006)

- Reiniciar la base de datos para que tome los cambios:

Figura 15. Instrucción para bajar la Base de Datos



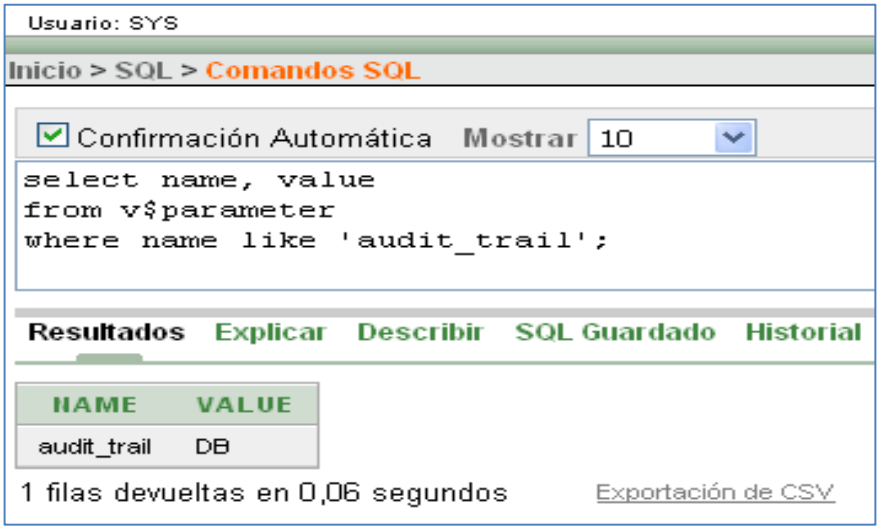
Fuente: Propia

Figura 16. Instrucción para subir la Base de Datos

Fuente: Propia

- Validar que haya tomado el cambio en el parámetro para habilitar la auditoria en la base de datos:

Figura 17. Validar activación de auditoria Oracle

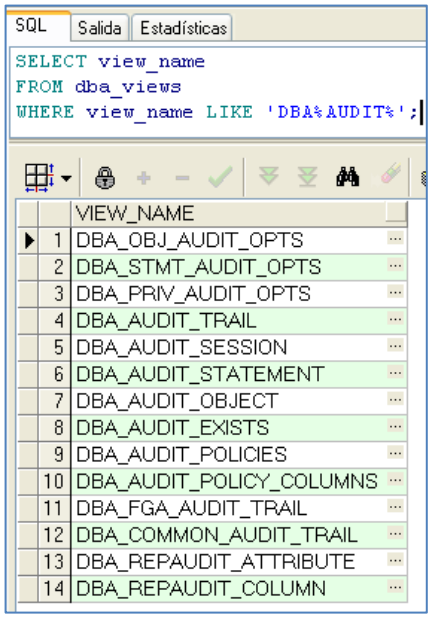


Fuente: Propia

Para detallar el contenido de la tabla $sys.aut, es posible hacerlo a través de las siguientes vistas de la base de datos:

SELECT view_name FROM dba_views WHERE view_name LIKE 'DBA%AUDIT%';

Figura 18. Consultar vistas de auditoria



Fuente: Propia

Oracle ofrece diferentes vistas de auditoria, basadas en la tabla SYS.AUD$, y FGA_LOG$, que ofrecen diferentes puntos de vista de los registros de auditoria de la base de datos, según la documentación oficial de Oracle [59]se encuentran las principales vistas de auditoria:

- DBA_AUDIT_OBJECT: Se muestra información detallada de la auditoría de objetos de la base de datos
- DBA_AUDIT_SESSION: Se muestra información detallada de la auditoría de los inicios de sesión de usuario.
- DBA_AUDIT_TRAIL: Se muestra información detallada de la auditoría estándar.
- USER_AUDIT_TRAIL: Se muestra información detallada de la auditoría estándar, del usuario actual. Todos los usuarios cuentan con esta vista.
- DBA_FGA_AUDIT_TRAIL: Se muestra información detallada de la auditoría de grano fino (FGA), obtenida de la tabla FGA_LOG$. La

[59] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Verifying Security Access with Auditing. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/network.112/e36292/auditing.htm#DBSEG006)

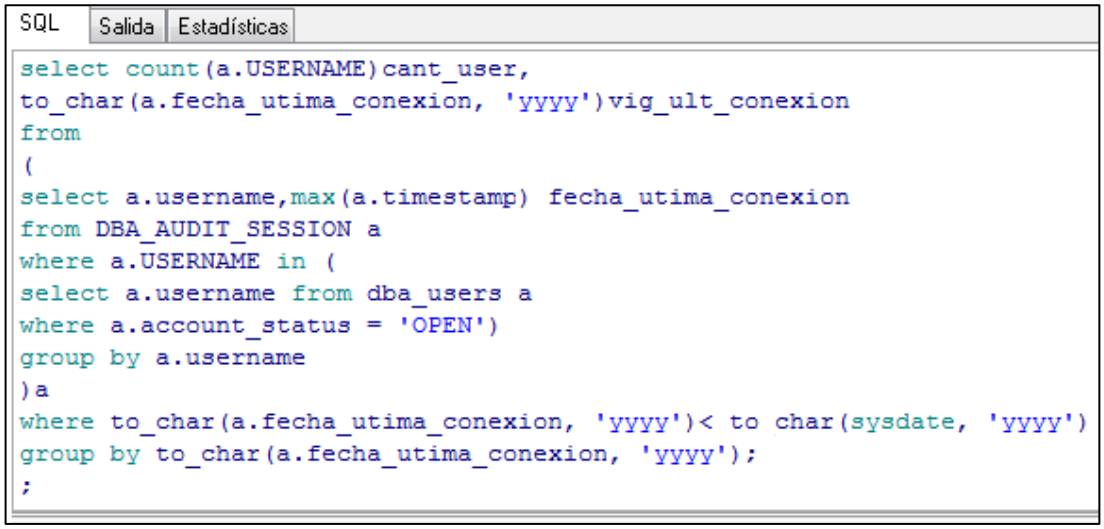
auditoría FGA extiende la auditoría estándar de conocer cual usuario ejecuta una determinada acción sobre un objeto, para conocer adicionalmente la auditoria a la sentencia sql que ejecuta el usuario sobre el objeto, los datos que fueron creados, borrados o actualizados por parte del usuario.

- **Validación de usuarios activos en desuso:** Dentro de las tablas de auditoria de Oracle, se encuentra la de auditoria de inicio de sesión de los usuarios, según la documentación oficial de Oracle[60], se encuentra la vista DBA_AUDIT_SESSION que es una vista útil para realizar un monitoreo a los usuarios activos de una base de datos pero que no hayan tenido actividad en ella en un tiempo determinado. El tiempo de inactividad prudente, lo define el dba o usuario encargado del monitoreo de la base de datos. Una consulta para identificar estos usuarios es la siguiente:

```
select count(a.USERNAME)cant_user,
to_char(a.fecha_utima_conexion, 'yyyy')vig_ult_conexion
from
(
select a.username,max(a.timestamp) fecha_utima_conexion
from DBA_AUDIT_SESSION a
where a.USERNAME in (
select a.username from dba_users a
where a.account_status = 'OPEN')
group by a.username
)a
where to_char(a.fecha_utima_conexion, 'yyyy')< to_char(sysdate, 'yyyy')
group by to_char(a.fecha_utima_conexion, 'yyyy');
```

[60] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_AUDIT_SESSION. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/server.112/e40402/statviews_3079.htm#REFRN23021)

Figura 19. Usuarios activos obsoletos en la base de datos



Fuente: Propia

Se toman todos los usuarios activos (dba_users) se comparan con los usuarios que ha realizados inicio de sesión en años anteriores al año actual (DBA_AUDIT_SESSION); Obteniendo el listado de los usuarios activos que no han realizado conexiones a la base de datos en el año actual, información que permite identificar a usuarios que deben ser pasadas a estado inactivo.

Al identificar los usuarios se debe proceder a bloquearlos, ya que pueden pertenecer a usuarios retirados de la organización o usuarios que ya no cuentan con actividades donde tengan que intervenir con información de la base de datos. Dentro de las buenas prácticas de seguridad en la organización se debe tener en cuenta incluir las tareas de bloqueo inmediato de personas retiradas de la organización y de reasignación de roles en la base de datos, cuando los usuarios son cambiados de funciones en la empresa.

8.2.4. Configuración de respaldo y restauración de la base de datos. Para garantizar la conservación de la información y la continuidad del negocio, se requiere contar con métodos y herramientas que permitan la pronta recuperación y minimicen la pérdida de datos; Para cumplir este objetivo se establecen políticas de seguridad y conservación de la información en las empresas; Oracle brinda opciones para la realización e backup físicos y lógicos de la base de datos como RECOVERY MANAGER (RMAN) y ORACLE DATA PUMP EXPORT/IMPORT, así como mecanismos para configurar la base de datos de tal forma que sea más óptima su restauración con ARCHIVELOG.

- **Configuración de modo archivelog en la base de datos**. Oracle cuenta con un mecanismo para respaldarse ante averías físicas del disco donde se guarda la base de datos, así como de modificaciones de los datos no autorizadas o no deseadas, ya sea por error de los usuarios, el administrador de la base de datos, o por debilidades en los aplicativos que interactúan con la base de datos o que no cuentan con suficientes controles para evitar que se realicen daños a los datos de forma involuntaria. Este mecanismo de respaldo se activa en la base de datos cuando es configurada en modo ARCHIVELOG, o modo de hacer copia a los archivos de redo log.

Según la documentación del proyecto AjpdSoft.Oracle dedicado a la publicación en internet de conocimientos en nuevas tecnologías, entre ellas base de datos, se describen ventajas que trae la configuración ARCHIVELOG en una base de datos Oracle, como la reducción de la posibilidad de pérdida de datos en caso de fallos, ya que hace posible que se puedan llevar a cabo restauraciones a la base de datos desde un punto específico de tiempo y permite realizar backup sin detener la base de datos (backup en caliente).

LA documentación oficial de Oracle[61] hace referencia a que defecto la base de datos se instalan en modo NOARCHIVELOG. Entre las desventajas que trae el contar con una base de datos en modo NOARCHIVELOG, está que en caso de fallo que requiera restaurar la base de datos, se perderían las transacciones que se hayan realizado desde el último backup hasta el momento de la restauración. Otra desventaja de este modo de configuración es que sólo es posible hacer copias de seguridad con la base de datos cerrada.

La configuración de la base de datos en modo ARCHIVELOG, es recomendable para base de datos que operan 24 horas 7 días a la semana y para cuando se manejan datos muy críticos donde la menor perdida de información puede ocasionar inconvenientes graves a la compañía, según lo detalla la documentación oficial de Oracle[62].

Según la documentación del proyecto de publicación en internet de conocimientos en nuevas tecnologías AjpdSoft.Oracle, específicamente en base de datos Oracle, detalla los siguientes pasos para configurar la base de datos en archivelog[63]:

- En una consola de Sql Plus, conectarse a la Base de datos que se va a validar: *connect usuario/contraseña@[Nombre_BD] as sysdba*

---

[61] (ORACLE CORPORATION. Oracle Help Cente, Database Administrator's Guide: Choosing Between NOARCHIVELOG and ARCHIVELOG Mode. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28310/archredo002.htm)

[62] Ibíd.

[63] (PROYECTO AjpdSoft, : Activar modo ARCHIVELOG en Oracle Database 11g R2 Bases de Datos. [En línea]. 2016, Disponible en: http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=559)

- Inicialmente, se debe conocer si la base de datos está en modo Archivelog. Existen varias formas de hacer esta validación, como las siguientes:

  Se ingresa la siguiente instrucción en una consola en SQL Plus:

  SQL> archive log list;

  Un ejemplo de lo que arrojaría la consulta si la base de datos estuviera en modo NOARCHIVELOG:

  Database log mode No Archive Mode
  Automatic archival Disabled
  Archive destination /oracle10/product/10.1.3/dbs/arch
  Oldest online log sequence 36
  Current log sequence 38

  Otra forma de validar el modo de configuración, es consultando en la vista de parámetros de oracle v$database, ejecutando la sentencia sql:

  select name, log_mode from v$database;

  Que arrojaría los posibles valores de *NOARCHIVELOG* o *ARCHIVELOG*

  También es posible validar el parámetro directamente en el archivo de configuración init.ora de la base de datos, donde el parámetro log_archive_start, debe estar en true

  log_archive_start = true

  - Bajar la base de datos: shutdown immediate
  - Montar la base de datos : startup mount
  - Ejecutar la sentencia: alter database archivelog
  - Abrir la base de datos: alter database open
  - Activar el archivado automático: alter system archive log start
  - shutdown immediate;

  Para validar se ejecuta nuevamente la sentencia: Archive log list ó *select log_mode from v$database;*

- **Respaldos y restauración físicos con recovery manager (rman).** Oracle permite la realización de respaldo y restauración de la base de datos a nivel físico mediante el uso de la Herramienta Recovery Manager (RMAN).

Según la documentación oficial de Oracle[64], entre las principales características de esta herramienta se encuentran:

- Permite programar la automatización de tareas para la generación de backup y recuperación, creando archivos de texto que contenga los comandos rman necesarios para ejecutar estos procesos, a su vez que se configuran con opciones del sistema operativo. De esta forma se convierte en una aliada importante para la ejecución de las estrategias de respaldo que se planteen en la organización.
- Permite configurar backups incrementales, permitiendo realizar copias solamente de los bloques que hayan cambiado respecto al último backup realizado. (sólo en modo ARCHIVELOG). este tipo de configuración permite realizar copias y restauración de forma más rápida, ocupando menos espacio en disco.
- Es posible ejecutarlo por medio de línea de comandos rman o por entorno gráfico con la herramienta Oracle Enterprise Manager que facilita su administración.
- Permite detectar bloques corruptos desde el proceso de respaldo. obteniendo información de las vistas V$BACKUP_CORRUPTION y V$COPY_CORRUPTION o V$DATABASE_BLOCK_CORRUPTION.
- Permite la realización de Backus en frio o en caliente , este último solo cuando la base de datos está configurada en modo ARCHIVELOG

Según la documentación oficial de Oracle[65], los comandos RMAN más comunes para ejecutar tareas de respaldo y restauración:

- Para comenzar a ejecutar comandos rman, desde una ventana de línea de comandos del sistema operativo se ejecuta 'rman', de esta forma el prompt cambia a RMAN>, y ya es posible enviar comandos propios de rman para ejecutar los diferentes procesos de respaldo y restauración de la base de datos.

Para conectarse a la base de datos desde rman:

- Con el usuario SYS de la base de datos:

RMAN> CONNECT TARGET SYS/pwd@prod

[64] (ORACLE CORPORATION.Oracle Help Centrer, Database Backup and Recovery User's Guide: 1 Introduction to Backup and Recovery [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRADV8001)

[65] (ORACLE CORPORATION.Oracle Help Centrer, Database Backup and Recovery User's Guide: 9 Backing Up the Database [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmbckba.htm#BRADV8003)

- Con el usuario del sistema operativo:

RMAN> CONNECT TARGET /

Entre los comandos RMAN más comunes a ejecutar están:

- Para conocer la configuración de rman en la base de datos:

RMAN> SHOW ALL;

- Copia de base de datos y archivos regdo logs, estando configurada la base de datos en modo ARCHIVELOG:

RMAN> BACKUP DATABASE PLUS ARCHIVELOG;

- Realizar un backup de la base de datos:

RMAN> BACKUP DATABASE;

- Realizar imagen del backup de todos los archivos de la base de datos.

RMAN> BACKUP AS COPY DATABASE;

- Para validar posibles backups corruptos incluyendo los redo log files.

BACKUP VALIDATE
DATABASE
ARCHIVELOG ALL;

- Visualizar bloques corruptos en las copias

SQL> SELECT * FROM V$DATABASE_BLOCK_CORRUPTION;

- Para listar las copias de la base de datos realizadas

RMAN> LIST BACKUP;
RMAN> LIST COPY;
RMAN>LIST BACKUP SUMMARY;

- Para salir de rman

RMAN> EXIT

- Un ejemplo de un archivo creado con alguna tarea de backup y su ejecución es el siguiente:

En un editor de texto se escribe:

 # archivo bkrman.txt
CONNECT TARGET /
BACKUP DATABASE PLUS ARCHIVELOG;
LIST BACKUP;
EXIT;

Para ejecutarlo, desde línea de comandos en el sistema operativo:

% rman @/my_dir/bkrman.txt

- **Respaldos y restauración lógicos con oracle data pump export/import.** Para realizar respaldos y restauración lógica de la base de datos, según la documentación oficial de Oracle[66], Oracle 11g cuenta con la funcionalidad de export/import datapump. Esta herramienta permite realizar backup lógicos ya sea de toda la base de datos, esquemas, u objetos específicos como tablas, procedimientos, paquetes, de una forma sencilla, creando unos archivos de fácil portabilidad. Sólo funciona con la base de datos abierta.

Para realizar respaldo con export datapump:

- ✓ Como Export FULL, se ejecuta la instrucción:
  expdp userid=$DBUSER/$DBPSWD dumpfile=$FILE.dmp logfile=$FILE.log full=yes
- ✓ Como export de un esquema, se ejecuta la instrucción:
  expdp userid=$DBUSER/$DBPSWD dumpfile=$FILE.dmp logfile=$FILE.log schemas=$SCHEMA

- **Configuración de políticas de respaldo.** Dependiendo de tamaño de BD y criticidad, se deben contar con planes de copias de seguridad y recuperación, que estén acorde a la continuidad que se espera del negocio. Un ejemplo de política podría ser:

- Respaldos con export datapump:

[66] (ORACLE CORPORATION.Oracle Help Centrer, Database Utilities: 2 Data Pump Export [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28319/dp_export.htm)

    - De Lunes a Viernes a las 12:30 PM
    - De Lunes a Viernes a las 6:45 PM
- Respaldos con RMAN:
    - Todos los días cada hora, backup de archive logs
    - Todos los días a las 11:30 PM, full backup

- **Políticas de seguridad relacionadas con respaldo y restauración.**
    - ✓ Dentro de la segregación de funciones en la Base de datos, se debe contar con un usuario que solo tenga privilegios para la generación de copias de seguridad y restauración. Esto debido a que las funciones de administración de la base de datos en una organización, en ocasiones se cuenta con varios responsables con diferentes funciones: monitoreo, gestión de usuarios, y encargado de las copias de seguridad.
    - ✓ Incluir en el plan de recuperación los pasos a seguir en caso de una incidencia critica en la BD.
    - ✓ Se debe contar con tareas de validación de los BK generados y cintas grabadas, para medir la calidad de los ficheros BK, y tiempos de respuesta ante un suceso de alerta.

Las cintas deben almacenarse en un lugar diferente a las instalaciones donde se encuentre el servidor de la base de dato

9. RESULTADO DE PRUEBAS A BASE DE DATOS DE LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC)

La Corporación Autónoma Regional del Valle del Cauca (CVC), autorizó llevar a cabo pruebas dentro de su entorno de desarrollo en las instalaciones de la entidad, sin tener acceso a producción. De igual forma sólo autoriza publicar parcialmente los resultados de las pruebas, debido a la criticidad de sus datos ya que son de índole estatal, no es permitido la divulgación de imágenes, ni código que exponga información propia de la Base de datos.

Las pruebas se realizaron en equipos ubicados dentro de las instalaciones de la CVC que contaban con herramientas instaladas para manipulación y ejecución de consultas sql en Oracle (SQL Developer y PL/SQL Developer), y una herramienta para escaneo de vulnerabilidad del servidor de la base de datos (Nessus).

Los equipos de cómputo asignados, contaban con acceso a la base de datos de pruebas, la cual es una copia espejo de la de la base de datos de producción que se encuentra configurada en entorno Oracle 11g. Para el caso de escaneo de vulnerabilidades del servidor que aloja la base de datos, se autorizó realizarlas al servidor donde se encuentra configurada la base de datos de producción, con restricción de la divulgación detallada de los resultados arrojados. Debido a que los aplicativos de la corporación no se encuentran configurados para acceder a la base de datos de desarrollo, adicional a la restricción del jefe del área para aplicar pruebas en ambiente de producción, no fue posible realizar test de vulnerabilidad a los aplicativos web de la entidad.

La práctica se lleva a cabo en una primera parte accediendo directamente a la base de datos de desarrollo con usuario y clave con rol de administrador y como segunda instancia utilizando herramienta de escaneo de vulnerabilidades del servidor donde está alojada la base de datos.

PRUEBAS CON ACCESO DIRECTO A LA BASE DE DATOS

Para las pruebas que se realizaron accediendo directamente a la base de datos, se siguieron los pasos y recomendaciones de la guía planteada en el desarrollo de este proyecto, ejecutando pruebas de la seguridad de la base de datos a nivel de:

- Control de acceso
- Protección de datos
- Políticas de respaldo y restauración

- Controles de auditoria

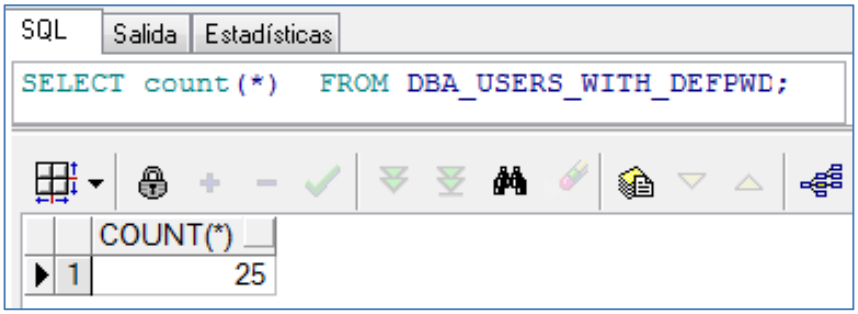
9.1 PRUEBAS DE CONTROL DE ACCESO

Se llevaron a cabo pruebas de control de acceso, para conocer: Usuarios con contraseñas creados por defecto en el proceso de instalación; Usuarios con privilegios de administrador; Usuarios con autenticación de sistema operativo o tipo de autenticación externa; Usuarios con privilegios del sistema. A continuación se detalla cada una de las pruebas realizadas.

- **Validación de usuarios y contraseñas creados por defecto:** Se ejecutó la consulta "SELECT * FROM DBA_USERS_WITH_DEFPWD", para validar usuarios creados en el proceso de instalación de la base de datos, que aun cuentan con la contraseña que se asigna por defecto. Para conocer la cantidad de usuarios se ejecutó la siguiente consulta sql:

SELECT count(*) FROM DBA_USERS_WITH_DEFPWD;

Figura 20- Prueba de validación usuarios por defecto



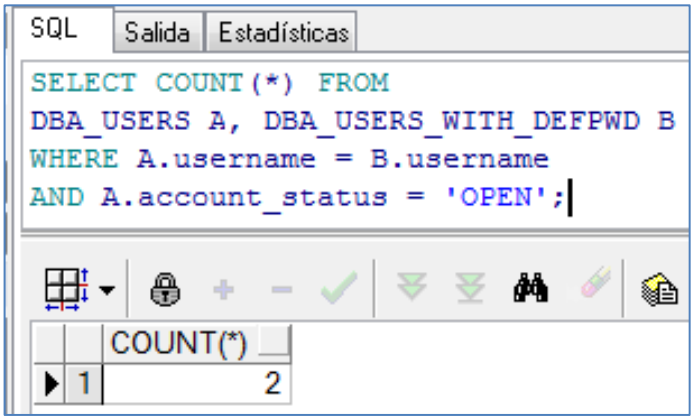
Fuente: Propia

Con la ejecución de la consulta se identificaron 25 cuentas de usuarios con esta característica.

Para validar de estos 25 usuarios creados por defecto durante la instalación, cuántos de ellos se encuentran aún activos, se hizo uso de la vista "DBA_USERS", donde se identificaron las cuentas de usuarios de la base de datos y el estado de la cuenta. Se generó el resultado a partir de la consulta sql:

SELECT COUNT(*) FROM DBA_USERS A, DBA_USERS_WITH_DEFPWD B WHERE A.username = B.username AND A.account_status = 'OPEN';

Figura 21 - Prueba de validación usuarios por defecto en estado abierto



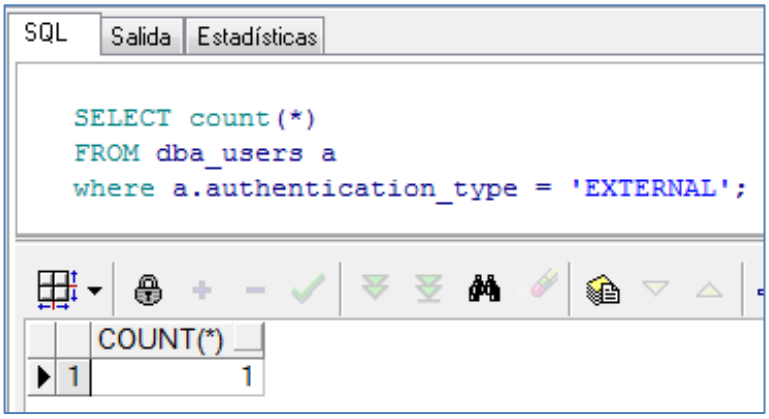
Fuente: Propia

Con la ejecución de la consulta anterior se identificaron 2 cuentas de usuario activas con clave por defecto de instalación.

- **Validación de usuario con autenticación de sistema operativo:** Para validar usuarios con tipo de autenticación “EXTERNAL”, es decir los usuarios que en este momento pueden acceder a la base de datos sin ingresar contraseña, normalmente el usuario de sistema operativo, se ejecutó la siguiente sentencia sql:

SELECT count(*) FROM dba_user A WHERE a.authentication_type = ‘EXTERNAL’;

Figura 22 - Prueba de validación usuarios con autenticación de SO

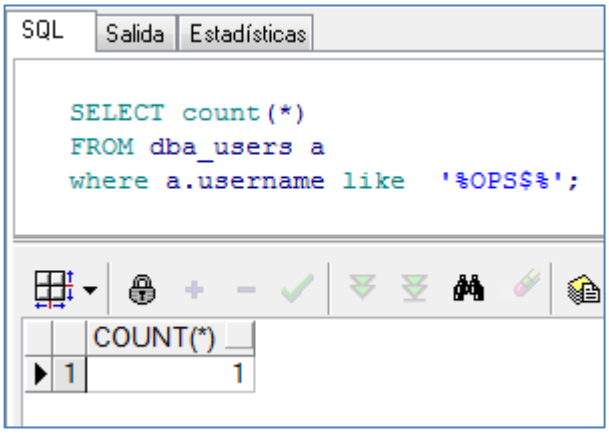


Fuente: Propia

Se encontró el mismo resultado ejecutando la sentencia sql para conocer usuarios con autenticación de sistema operativo, que son creados con el prefijo ‘OPS$’:

*SELECT * FROM dba_users a where a.username like '%OPS$%'*

Figura 23 - Prueba de validación usuarios con autenticación externa

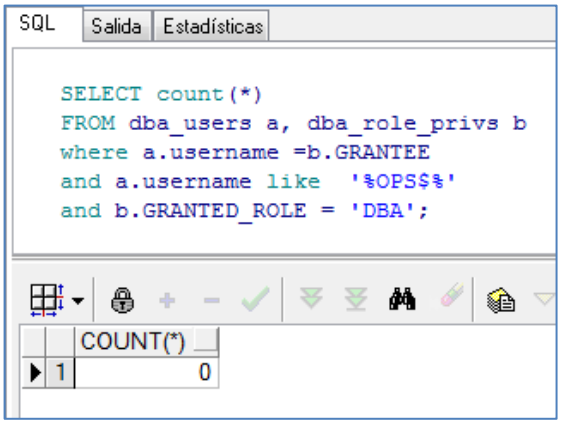


Fuente: Propia

Con la ejecución de cualquiera de las dos consultas anteriores se identificó 1 cuenta de usuario con esta característica. Para conocer si este usuario de sistema operativo, cuenta con privilegios de administrador se ejecutó la siguiente sentencia sql:

SELECT count(*) FROM dba_users a, dba_role_privs b where a.username = b.GRANTEE and a.username like '%OPS$%' and b.GRANTED_ROLE = 'DBA';

Figura 24 - Prueba de validación usuarios de SO con permisos DBA



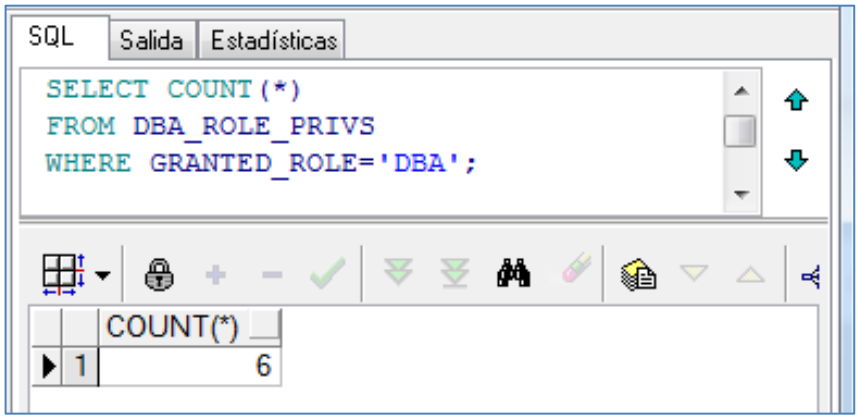
Fuente: Propia

Se encuentra que el usuario del sistema operativo que tiene acceso directo a la base de datos, no cuenta con privilegios de administrador.

- **Validación de usuario con privilegios de administrador:** Se ejecutó la siguiente sentencia sql para validar el número de usuarios con rol de administrador:

SELECT COUNT(*) FROM DBA_ROLE_PRIVS WHERE GRANTED_ROLE='DBA';

Figura 25 - Prueba de validación usuarios con permisos DBA



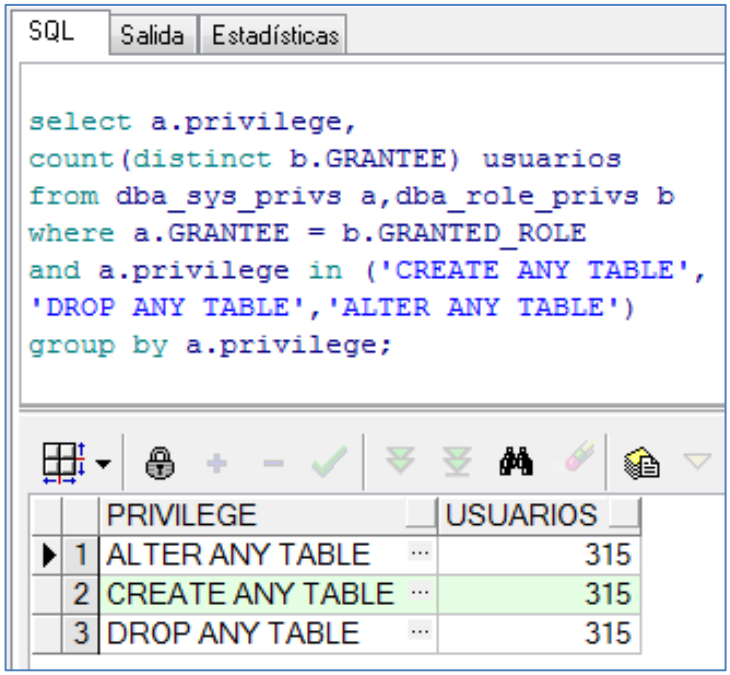
Fuente: Propia

Como resultado de la ejecución de la sentencia sql anterior, se encontró en total 6 usuarios con privilegios de administrador.

- **Validación de usuarios con privilegios para ejecutar comandos DDL:** Para conocer la cantidad de usuarios con permisos en el sistema para ejecutar comandos DDL (data definition language), como CREATE, ALTER y DROP, sobre todas las tablas, se ejecutó la siguiente sentencia sql:

```
select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by a.privilege;
```

Figura 26 - Prueba de validación usuarios con permisos DDL

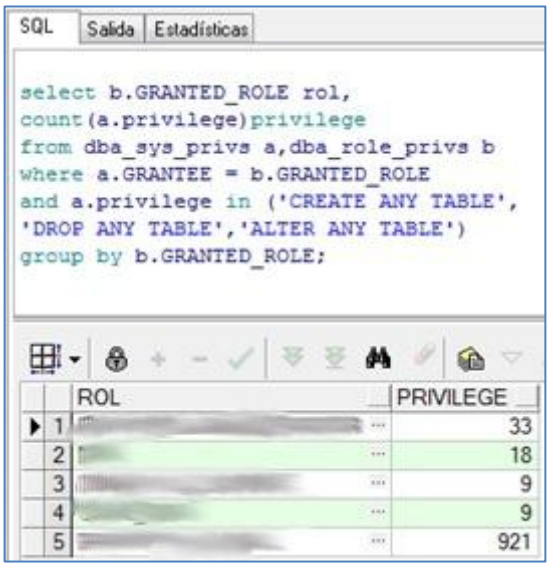


Fuente: Propia

Debido a que se encuentra un número alto de usuarios con privilegios para ejecutar comandos DDL, se procedió a identificar cual era el principal rol de la base de datos que cuenta con estos privilegios que se le asignan a los usuarios. Se ejecutó la siguiente consulta sql:

select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by b.GRANTED_ROLE;

Figura 27 - Prueba de validación roles con permisos DDL



Fuente: Propia

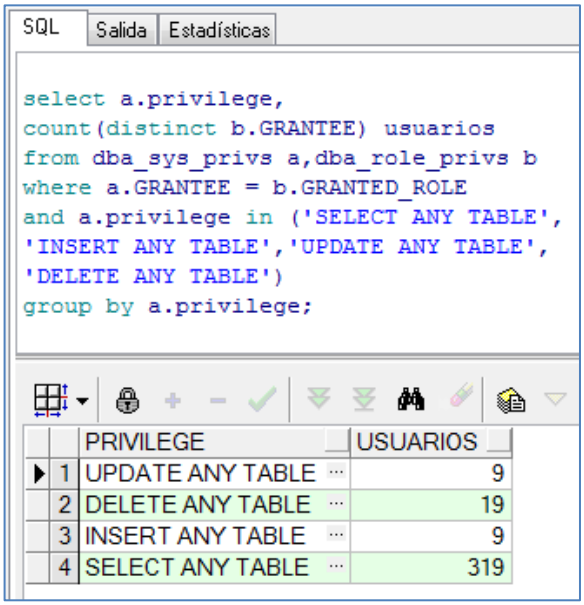
Como resultado de la ejecución de la consulta anterior, se identifica un rol de la base de datos que ha sido asignado a un número alto de usuarios otorgándoles privilegios para crear, modificar o borrar la estructura de objetos de la base de datos.

- **Validación de usuarios con privilegios DML sobre objetos del sistema:**

Para conocer la cantidad de usuarios con permisos en el sistema para ejecutar comandos DML (Data manipulation language), como SELECT, INSERT, UPDATE y DELETE sobre todas las tablas, se ejecutaron las siguientes sentencias sql:

```
select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by a.privilege;
```

Figura 28 - Prueba de validación usuarios con permisos DML

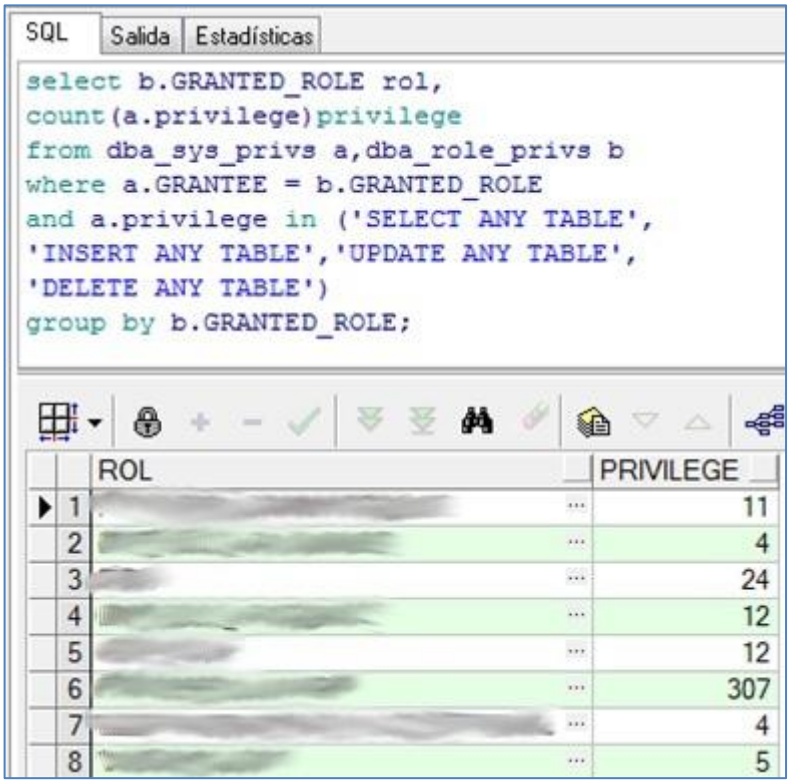


Fuente: Propia

Debido a que se encuentra un número alto de usuarios con privilegios para ejecutar comandos DML, se procedió a identificar cual era el principal rol de la base de datos que cuenta con estos privilegios que se le asignan a los usuarios. Se ejecutó la siguiente consulta sql:

select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by b.GRANTED_ROLE;

Figura 29 - Prueba de validación roles con permisos DML

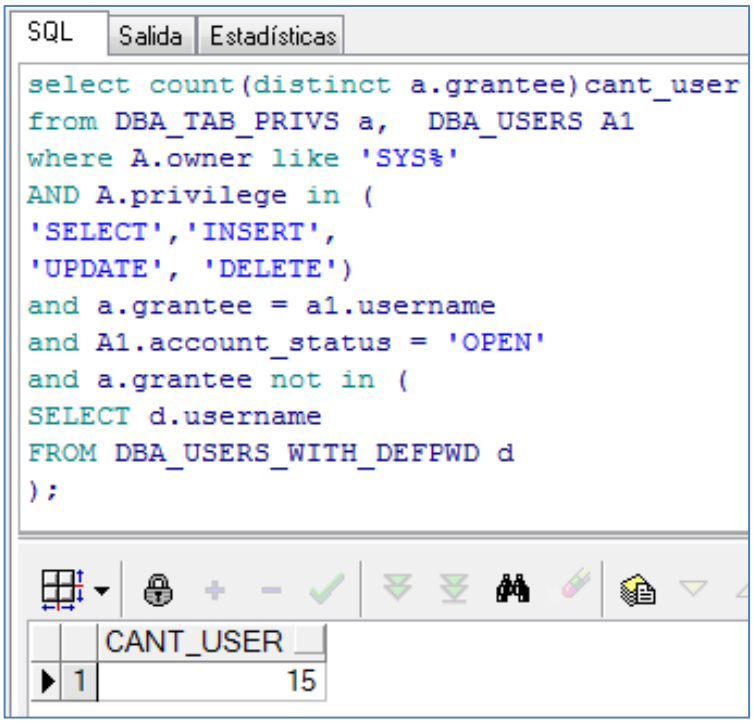


Fuente: Propia

Otra forma que se utilizó para conocer la cantidad de usuarios que cuentan con privilegios para ejecutar comandos DML en objetos, cuyo propietario es el usuario SYS fue usando la vista DBA_TAB_PRIVS ejecutando la siguiente sentencia sql:

```
select count(distinct a.grantee)cant_user
from DBA_TAB_PRIVS a,  DBA_USERS A1
where A.owner like 'SYS%'
AND A.privilege in (
'SELECT','INSERT',
'UPDATE', 'DELETE')
and a.grantee = a1.username
and A1.account_status = 'OPEN'
and a.grantee not in (
SELECT d.username
FROM DBA_USERS_WITH_DEFPWD d );
```

Figura 30 - Prueba de validación roles con permisos DML sobre objetos del sistema



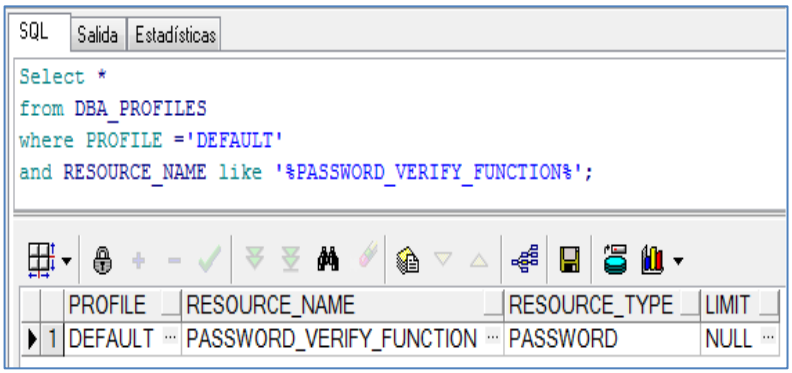
Fuente: Propia

Se encontraron en total 15 usuarios con privilegios para ejecutar comandos DML en objetos del sistema.

- **Políticas de contraseñas:** Para conocer la complejidad en la construcción de las contraseñas, se valida en la base de datos la configuración del perfil que se asigna en la creación de usuarios. El perfil asignado a los usuarios en la creación es 'DEFAULT'. De esta forma se ejecuta la siguiente consulta para conocer si el perfil cuenta con algún valor asignado la característica de PASSWORD_VERIFY_FUNCTION:

```
Select *
from DBA_PROFILES
where PROFILE ='DEFAULT'
and RESOURCE_NAME like '%PASSWORD_VERIFY_FUNCTION%';
```

Figura 31 - Validación de políticas de contraseñas



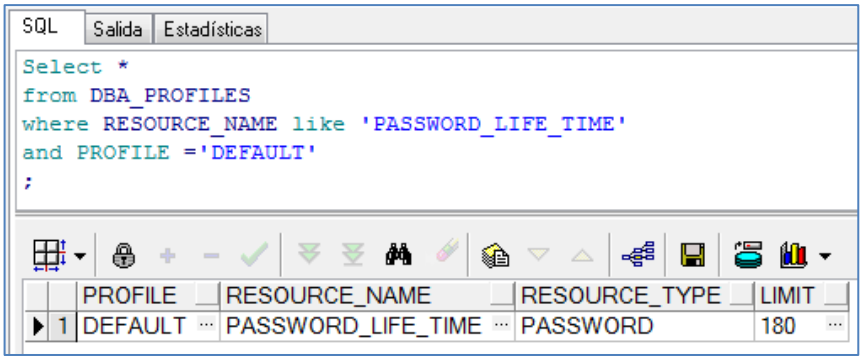
Fuente: Propia

Se encuentra que el profile 'DEFAULT' en su característica '*PASSWORD_VERIFY_FUNCTION',* se encuentra asignado un valor nulo, lo que indica que no cuentan con ninguna función que otorgue seguridad en la complejidad exigida para la construcción de las contraseñas de los usuarios de la base de datos.

Otra consulta a tener en cuenta es la de las características de tiempo en días de caducidad de la contraseña, ejecutando la consulta sql:

*Select **
from DBA_PROFILES
where RESOURCE_NAME like 'PASSWORD_LIFE_TIME'
and PROFILE ='DEFAULT';

Figura 32 - Validación de caducidad de contraseñas



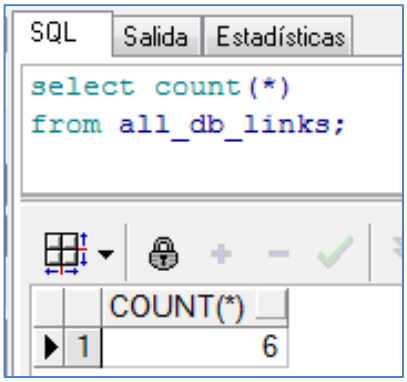
Fuente: Propia

Se encuentra que si se tiene configurado un límite de días, para la caducidad de la contraseña de los usuarios de la base de datos.

- **Acceso restringido a información de base de datos remotas:** Para conocer si dentro de la base de datos se cuenta con buenas prácticas de creación de dblink para conectarse a otras base de datos, se ejecuta la siguiente sentencia sql:

select count(*) from DBA_DB_LINKS;

Figura 33 - Validación de DBlinks



Fuente: Propia

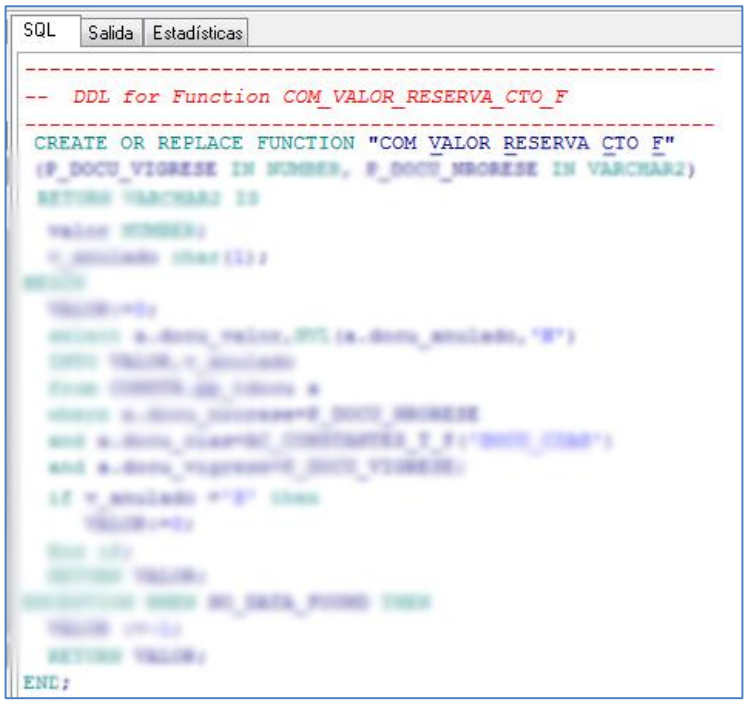
Se encontraron en total 6 dblink construidos en la base de datos.

9.2. PRUEBAS DE PROTECCIÓN DE DATOS

Se llevaron a cabo pruebas de protección de los datos, para conocer la protección de objetos construidos con lenguaje pl/sql, almacenado en la base de datos. A continuación se detallan las pruebas realizadas a este nivel.

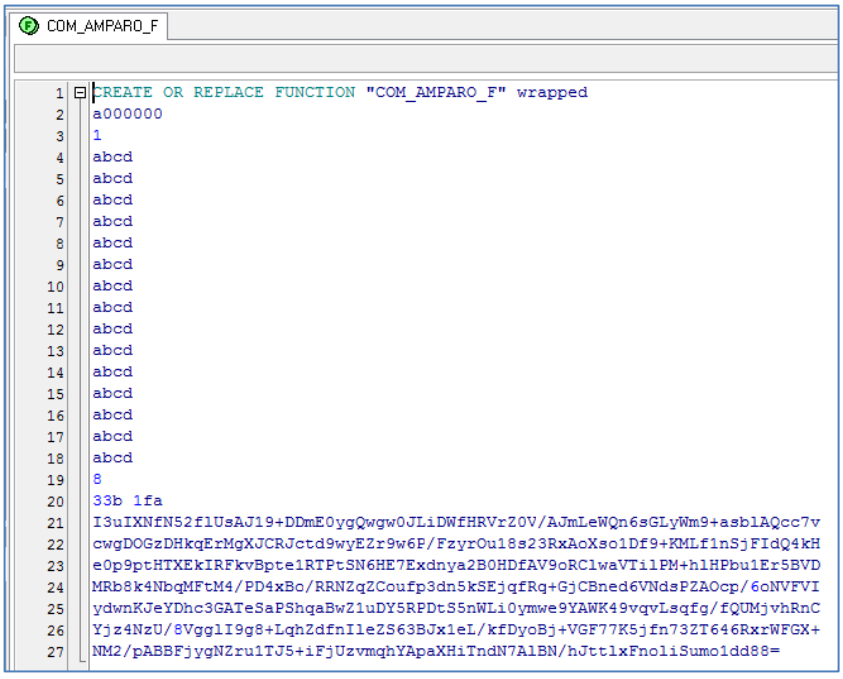
- **Validación de protección de objetos de la base de datos:** Se llevaron a cabo consultas directas a objetos de la base de datos construidos con lenguaje pl/sql, identificados como críticos del negocio dentro de la base de datos. A través de bitácoras de los objetos se identificaron y se seleccionaron aleatoriamente objetos para conocer si contaban con protección a nivel de encriptación.

Figura 34 - Validación de encriptación de objetos



Fuente: Propia

Figura 35 - Validación de objetos PL/SQL encriptados



Fuente: Propia

Se encontró objetos que almacenan código PL con lógica crítica del negocio, que aún no se encuentra encriptado.

9.3. PRUEBAS DE CONFIGURACIÓN DE MECANISMOS DE AUDITORIA

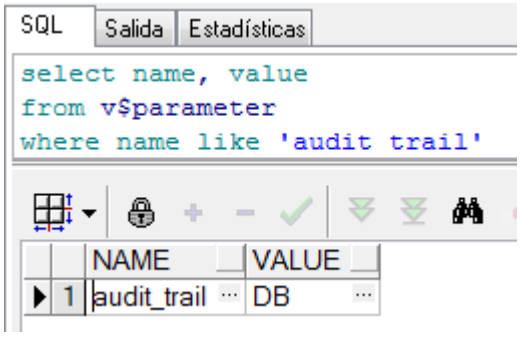
Se llevaron a cabo pruebas para conocer si se cuentan con configuraciones de mecanismos de auditoria. A continuación se detallan las pruebas realizadas a este nivel.

- **Validación de Estado de activación de auditoria de la Base de datos:**

Para validar si la instancia de la Base de datos tiene activa la configuración de auditoria, se ejecutó la siguiente instrucción SQL:

select name, value from v$parameter where name like 'audit_trail'

Figura 36 - Validación de estado de activación de auditoria de la BD



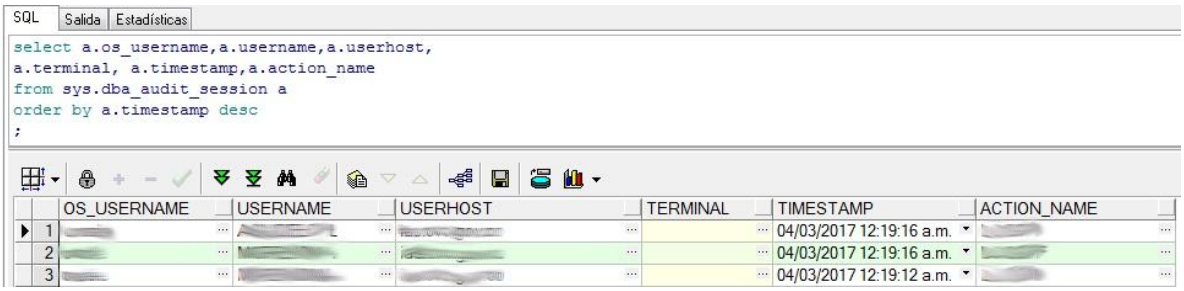
Fuente: Propia

Se evidencia que se encuentra activa la configuración de auditoria, de tal forma que se está almacenando en SYS.AUD$

Se lleva cabo la validación del registro de la auditoria de la base de datos, consultando dos de las principales vistas: DBA_AUDIT_SESSION y DBA_AUDIT_TRAIL, ejecutando las siguientes sentencias SQL:

```
Select a.os_username,a.username,a.userhost,
a.terminal, a.timestamp,a.action_name
from sys.dba_audit_session a
order by a.timestamp desc
```

Figura 37 - Auditoria de sesión de usuario

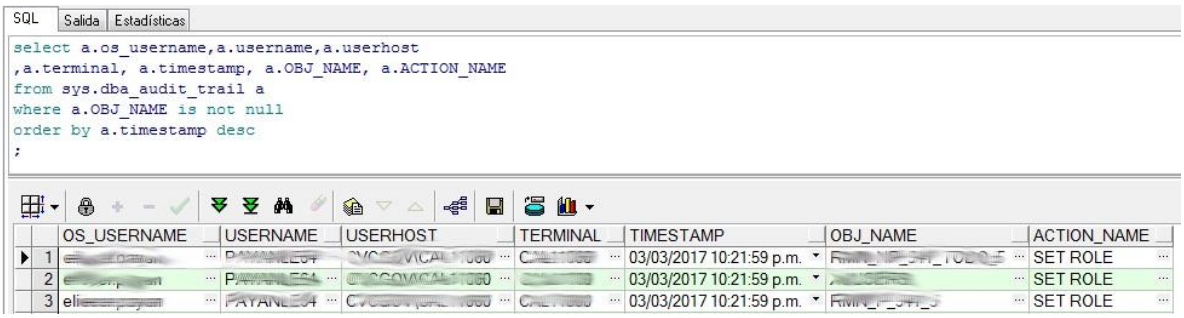


Fuente: Propia

Con la anterior instrucción se validó que se estuviera registrando auditoria de sesión de usuarios, evidenciando la acción de ingreso y salida de la base de datos y la fecha y hora, entre otros datos relevantes.

select a.os_username,a.username,a.userhost
,a.terminal, a.timestamp, a.OBJ_NAME, a.ACTION_NAME
from sys.dba_audit_trail a
where a.OBJ_NAME is not null
order by a.timestamp desc

Figura 38 - Auditoria de acciones sobre objetos



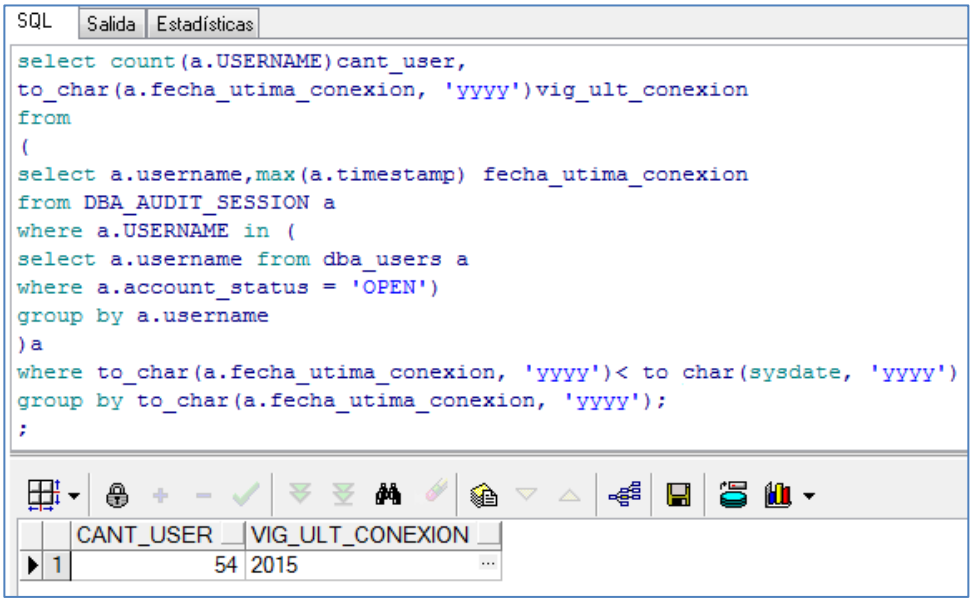
Fuente: Propia

Con la anterior instrucción se validó que se estuviera registrando auditoria sobre las acciones realizadas a los objetos de la base de datos, evidenciando el nombre del objeto, usuario, terminal y acción realizada entre otra información relevante.

- **Validación de usuarios activos en desuso:** Validación de usuarios activos en desuso: Para identificar usuarios activos en desuso, se ejecutó la siguiente sentencia sql:

```
select count(a.USERNAME)cant_user,
to_char(a.fecha_utima_conexion, 'yyyy')vig_ult_conexion
from
(
select a.username,max(a.timestamp) fecha_utima_conexion
from DBA_AUDIT_SESSION a
where a.USERNAME in (
select a.username from dba_users a
where a.account_status = 'OPEN')
group by a.username
)a
where to_char(a.fecha_utima_conexion, 'yyyy')< to_char(sysdate, 'yyyy')
group by to_char(a.fecha_utima_conexion, 'yyyy');
```

Figura 39 - Validación de usuarios en desuso



Fuente: Propia

Para la consulta se tuvo en cuenta la auditoria de sesiones de usuario, teniendo en cuenta que el último acceso realizado por los usuarios activos haya sido el año anterior. Se encontraron en total 54 usuarios activos, los cuales su último acceso a la base de datos corresponde al año anterior.

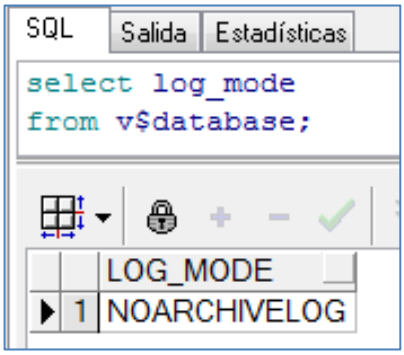
9.4. PRUEBAS DE CONFIGURACIÓN DE RESPALDO Y RESTAURACIÓN

Se llevaron a cabo pruebas para conocer si se cuentan con configuraciones de mecanismos de respaldo. A continuación se detallan las pruebas realizadas a este nivel.

- Validación de la configuración de la base de datos en modo Archivelog: Para conocer si la base de datos se encuentra configurada en modo Archivelog, se ejecutó la siguiente sentencia SQL:

  select log_mode from v$database;

Figura 40 - Validación de estado ARCHIVELOG de la BD



Fuente: Propia

Se encontró que la base de datos se encuentra configurada como NOARCHILOG.

9.5. POLÍTICAS DE RESPALDO

En entrevista con el actual administrador de la base de datos de la empresa, se confirma que existe un plan de políticas de respaldo de la base de datos y que se encuentra operando actualmente mediante tareas automáticas que se ejecutan según la prioridad establecida en el plan. De igual forma se confirma que se encuentra creado un usuario con permisos exclusivos para que ejecute la función de crear el respaldo de la base de datos, diferente al usuario sys. Por motivos de seguridad, la empresa no autoriza divulgar el plan de respaldo y restauración, ni scripts utilizados para la ejecución automática y periódica del respaldo de la base de datos. En la entrevista con el dba, se pudo confirmar que existe falencias es en la ejecución de las tareas de validación periódicas de las copias de respaldo

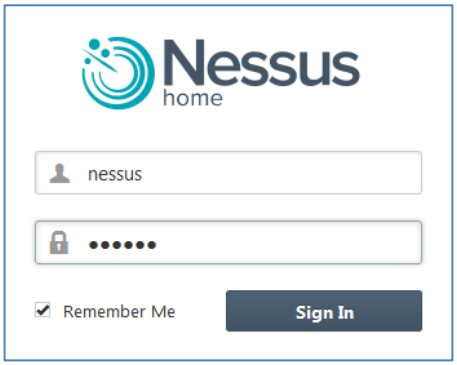
creadas a diario, por la falta de continuidad en la contratación de la persona encargada de la administración de la base de datos.

ESCANEO DE VULNERABILIDADES DE LA BASE DE DATOS

El escaneo de vulnerabilidades del servidor de la base de datos se lleva a cabo con la herramienta Nessus-6.9.1 para sistema operativo Windows. La IP del servidor fue suministrada directamente por el administrador de la base de datos de la empresa.

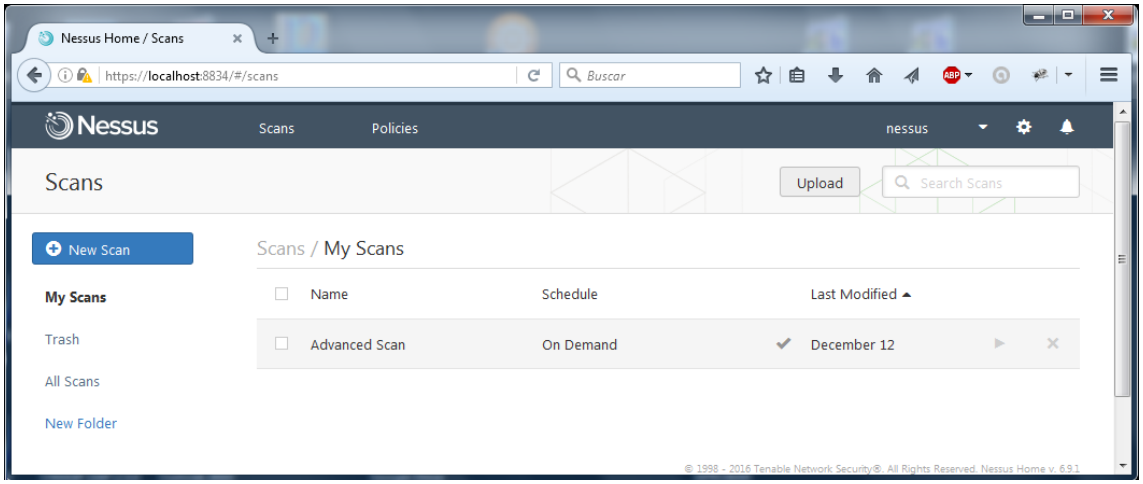
Se realizan los pasos necesarios para la activación del producto y creación del usuario con el que se va ingresar a la herramienta.

Figura 41- Conexión a la herramienta de escaneo Nessus



Fuente: Propia

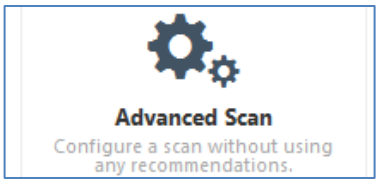
Figura 42 - Pantalla principal de la herramienta Nessus



Fuente: Propia

En la herramienta Nessus, se configuró una política de escaneo avanzado (Advanced Scan), para ejecutar la tarea.

Figura 43 - Tipo de política de escaneo de vulnerabilidad utilizada

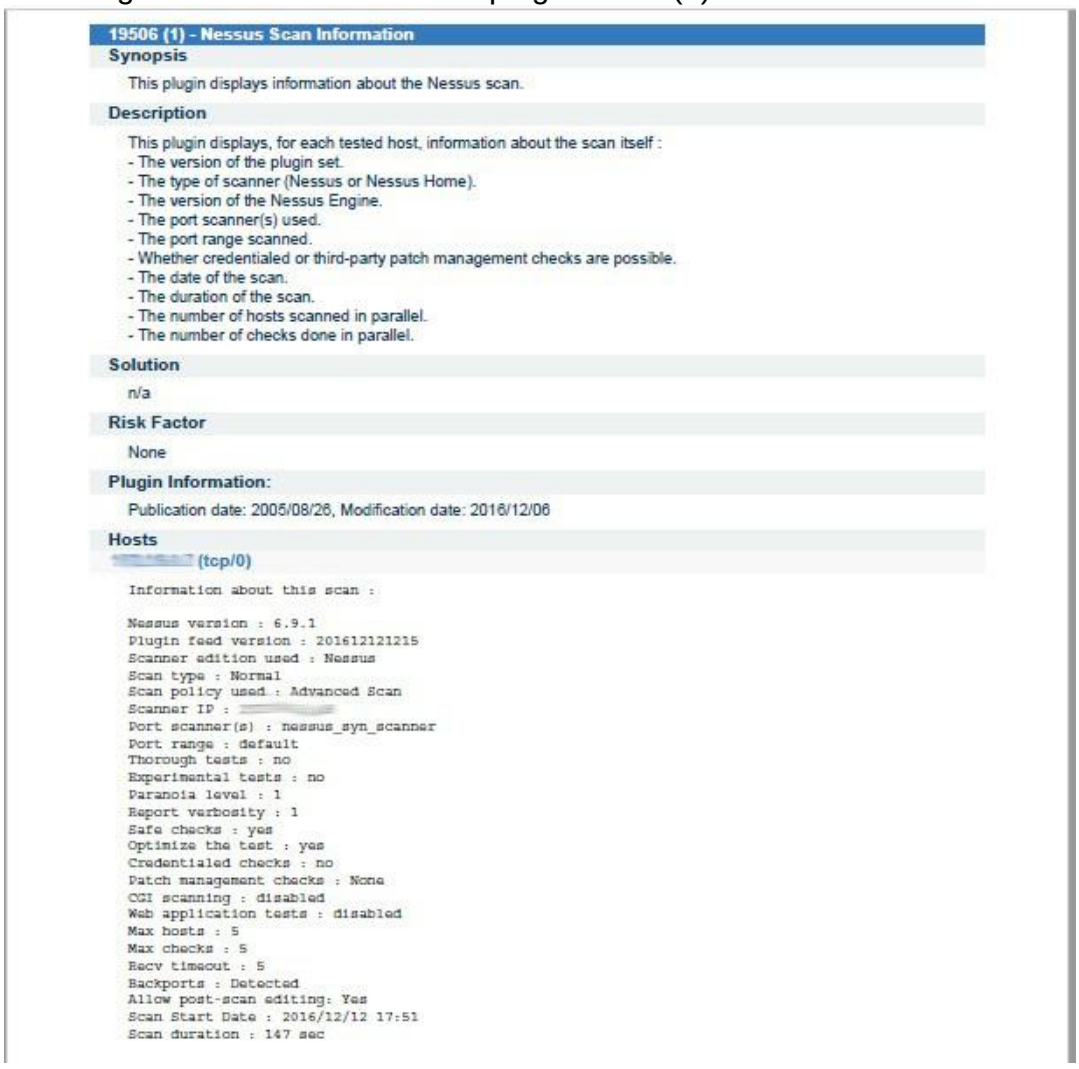


Fuente: Propia

En Nessus para llevar a cabo el chequeo de vulnerabilidades, se debe tener en cuenta que la herramienta cuenta con una lista de plugins, que se tienen como como referencia para ejecutar el escaneo.

Al finalizar el escaneo de vulnerabilidades, se genera una serie de informes sobre los resultados obtenidos, clasificándolos por nivel de criticidad de la vulnerabilidad encontrada y familia de plugin al que hace referencia. Entre los informes se muestra el resultado del plugin '9506 (1) - Nessus Scan Information', que detalla información propia de la herramienta como versión, tipo de configuración de la política de escaneo utilizada y duración.

Figura 44 - el resultado del plugin 9506 (1) - Nessus Scan Information



19506 (1) - Nessus Scan Information

Synopsis

This plugin displays information about the Nessus scan.

Description

This plugin displays, for each tested host, information about the scan itself :
- The version of the plugin set.
- The type of scanner (Nessus or Nessus Home).
- The version of the Nessus Engine.
- The port scanner(s) used.
- The port range scanned.
- Whether credentialed or third-party patch management checks are possible.
- The date of the scan.
- The duration of the scan.
- The number of hosts scanned in parallel.
- The number of checks done in parallel.

Solution

n/a

Risk Factor

None

Plugin Information:

Publication date: 2005/08/26, Modification date: 2016/12/06

Hosts

[redacted] (tcp/0)

```
Information about this scan :

Nessus version : 6.9.1
Plugin feed version : 201612121215
Scanner edition used : Nessus
Scan type : Normal
Scan policy used : Advanced Scan
Scanner IP : [redacted]
Port scanner(s) : nessus_syn_scanner
Port range : default
Thorough tests : no
Experimental tests : no
Paranoia level : 1
Report verbosity : 1
Safe checks : yes
Optimize the test : yes
Credentialed checks : no
Patch management checks : None
CGI scanning : disabled
Web application tests : disabled
Max hosts : 5
Max checks : 5
Recv timeout : 5
Backports : Detected
Allow post-scan editing: Yes
Scan Start Date : 2016/12/12 17:51
Scan duration : 147 sec
```

Fuente: Propia

La descripción general de las vulnerabilidades que arrojó el escaneo al servidor de la base de datos, se muestran en convenciones de colores para identificar el nivel de vulnerabilidad.

Figura 45 - Resultado de vulnerabilidades

Summary

Critical	High	Medium	Low	Info	Total
0	1	1	3	21	26

Details

Severity	Plugin Id	Name
High (7.5)	69552	Oracle TNS Listener Remote Poisoning
Medium (4.3)	90317	SSH Weak Algorithms Supported
Low (3.3)	11197	Multiple Ethernet Driver Frame Padding Information Disclosure (Etherleak)
Low (2.6)	70658	SSH Server CBC Mode Ciphers Enabled
Low (2.6)	71049	SSH Weak MAC Algorithms Enabled

Fuente: Propia

Se identifican en total 5 vulnerabilidades, dos de ellas de especial cuidado que se detallan a continuación:

Vulnerabilidad Alta:

- 69552 Oracle TNS Listener Remote Poisoning

Este resultado arroja una alerta sobre el registro de comunicación que se permite de hosts remotos al servidor de base de datos. Esta vulnerabilidad puede ser explotada remotamente, sin ningún tipo de autenticación en la base de datos, pudiendo afectar la confidencialidad, integridad y disponibilidad de los datos. Permite que un atacante pueda manipular la instancia de la base de datos, controlando el tráfico entre servidor-cliente desviando información del servidor legitimo a otro servidor. Se puede ocasionar ataques man-in-the-middle, secuestro de sesión (session- hijacking), ataques de denegación de servicio en el servidor que alberga la base de datos y podría realizar envío de comandos para elevación de privilegios.

El informe del escaneo, dispone de información complementaria al hallazgo de esta vulnerabilidad en la sección ‘66334 (1) - Patch Report’, donde se identifica que el servidor no cuenta con todos los parches de seguridad necesarios, identificando en esta misma sección la vulnerabilidad ‘Oracle TNS Listener Remote Poisoning’.

Figura 46 - Resultado de plugin 66334 (1) - Patch Report

Fuente: Propia

Dentro de las soluciones que se muestran en el informe que arroja la herramienta Nesuus para esta vulnerabilidad, está el realizar tareas basadas en la documentación y asesoramiento de Oracle, en cuanto a actualización de parches de seguridad que ya ha liberado la empresa para eliminar esta vulnerabilidad.

Vulnerabilidad Media:

- 90317 SSH Weak Algorithms Supported

Esta vulnerabilidad arrojada, alerta sobre los débiles algoritmos de encriptación utilizados en el servidor remoto SSH. El servidor SSH remoto está configurado para permitir algoritmos de cifrado RC4 el cual esta desactualizado y del cual se ha demostrado sus vulnerabilidades en el cifrado de información.

Dentro de las soluciones que se muestran en el informe que arroja la herramienta Nesuus para esta vulnerabilidad, se recomienda cambiarlo o reemplazar cualquier algoritmo de cifrado de flujo a algoritmos de cifrado en bloque. Para mitigar esta vulnerabilidad se aconseja la actualización de los navegadores de internet a las versiones más recientes, de las cuales se recomienda el uso de las versiones Chrome 30, Firefox 28, Internet Explorer 11 o superiores versiones que no usan el algoritmo RC4.

10. RESULTADOS Y DISCUSIÓN

Una vez se obtienen los resultados de las pruebas de configuración de seguridad de la Base de datos de la Corporación Autónoma Regional del Valle del Cauca (CVC). Se llevan a cabo una lista de recomendaciones al área de Informática para ser socializadas en conjunto con el administrador de la base de datos, clasificándolas en los diferentes niveles de seguridad abarcados en las pruebas:

- Control de acceso
- Protección de datos
- Políticas de respaldo y restauración
- Controles de auditoria
- A nivel de servidor de base de datos

Las recomendaciones generales que se plantean para cada nivel de seguridad abarcado son las siguientes:

10.1. CONTROL DE ACCESO

- **Usuarios y contraseñas creados por defecto:** Se debe validar el uso y privilegios que tienen asignados los usuarios encontrados en esta prueba realizada en la base de datos, que contaban con nombre y contraseña asignados por defecto en la instalación (dos usuarios en total), para considerar bloquearlos o si están siendo usados cambiar las claves que le fueron asignadas por defecto. Esta acción es necesaria llevarla a cabo, ya que los usuarios y claves que se crean por defecto durante el proceso de instalación de la base de datos Oracle, son de fácil conocimiento público, y ponen en riesgo la seguridad de la información ya que puede ser usada por un atacante para ejecutar acciones malintencionadas. .Para solventar este punto, es posible seguir las recomendaciones de la guía propuesta en el proyecto en punto de Control de acceso, en la sección: Revisión de usuarios por defecto.

- **Usuarios con autenticación de sistema operativo:** Se encontró un usuario con autenticación de sistema operativo, pero que no cuenta con privilegios de administración. Debe validarse de igual forma los privilegios del usuario con autenticación de sistema operativo para que solo cuente con los privilegios mínimos necesarios. De igual forma dentro de las tareas periódicas de seguridad en la base de datos se recomienda realizar este tipo de validación para monitorear que este usuario se encuentre siempre con privilegios restringidos. Como referencia para llevar a cabo controles sobre este punto de la seguridad de la base de datos, es posible seguir las

recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección: Usuarios con autenticación de sistema operativo.

- **Usuarios con privilegios de administración:** Debe tenerse en consideración, si la cantidad de usuarios encontrados en las pruebas (6 en total) son los que deben contar con los privilegios elevados, esta información debe ser de conocimiento del actual administrador de la base de datos de la empresa. La ejecución de esta sentencia periódicamente, permite buscar inconsistencias en el número de usuarios con privilegios DBA, que posiblemente hayan sido creados por un atacante. Como referencia para llevar a cabo controles sobre este punto de la seguridad de la base de datos, es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección: Usuario con privilegios de administración

- **Usuarios con privilegios para ejecutar comandos DDL:** Las pruebas identificaron un rol que es asignado a un número alto de usuarios, y que cuenta con privilegios que permiten crear, borrar o alterar tablas. Debido a que no es común que en una base de datos se encuentre un número elevado de usuarios con este tipo de privilegios, ya que podrían realizar acciones a objetos de la base de datos sin ninguna restricción. Se recomienda eliminar estos privilegios del rol. Como referencia para llevar a cabo controles sobre este punto de la seguridad de la base de datos, es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección: Usuarios con privilegios para ejecutar comandos DDL.

- **Usuarios con privilegios DML sobre objetos del sistema:** Debe tenerse en consideración, si la cantidad de usuarios encontrados son los que deben contar con los privilegios para ejecutar comandos DML sobre objetos del sistema cuyo dueño es el usuario SYS. esta información debe ser de conocimiento del actual administrador de la base de datos de la empresa. La ejecución de esta sentencia periódicamente, permite buscar inconsistencias en el número de usuarios con privilegios sobre objetos del sistema, ya que posiblemente hayan sido creados usuarios o haberse otorgado privilegios por parte de un atacante. Como referencia para llevar a cabo controles sobre este punto de la seguridad de la base de datos, es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección: Usuarios con privilegios DML sobre objetos del sistema

- **Políticas de contraseñas:** Debe tenerse en consideración, determinar una función en el profile utilizado en la creación de usuarios, para validar la complejidad de construcción de las contraseñas aplicando unas políticas de

seguridad más robustas. Para llevar a cabo esta tarea es posible hacer uso de la función que ya trae la base de datos Oracle 11g, o personalizar los controles a utilizar creando alguna función en la base de datos que exija algún tipo de restricción, como: contar con una longitud mínima, no permitir la asignación el nombre del usuario como clave, obligatoriedad del uso de un mínimo de caracteres especiales y letras mayúsculas. Este tipo de controles de seguridad en la complejidad de la contraseña, permitirá que no se facilite la tarea de un atacante, para acceder a la base de datos, ya que se disminuiría la vulnerabilidad de las contraseñas al no ser tan sencillo obtenerlas. Para solventar este punto, es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección Políticas de contraseñas

10.2. PROTECCIÓN DE DATOS

- **Validación de protección de objetos de la base de datos:** Se debe unificar las políticas de seguridad para los objetos de la base de datos en todos los esquemas, de tal forma que los objetos de la base de datos que manejan los principales procesos o datos del negocio se encuentren protegidos por mecanismo de encriptación. Para llevar a cabo este proceso, se requiere inicialmente realizar tareas para identificar objetos críticos de la base de datos que necesita protegerse, ejecutando consultas de auditoria y solicitando directamente a los proveedores de los aplicativos corporativos que listen los objetos. Para solventar este punto, es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección Encriptación a nivel de código PL/SQL almacenada en la Base de Datos.

10.3. CONTROLES DE AUDITORIA

- **Validación de usuarios activos en desuso** Se recomienda llevar a cabo tareas periódicas de monitoreo de la base de datos para identificar usuarios en desuso, teniendo en cuenta últimos accesos a la base de datos presentados por las auditorias, para proceder a su eliminación o bloqueo. En el caso específico de la prueba realizada, donde se encontraron 54 usuarios activos de la base de datos que no han accedido al sistema en el presente año, se recomienda inactivarlos, ya que pueden pertenecer a personas que ya no estén laborando en la empresa y como buena práctica se debe contar con procedimientos claros cuando hay retiros de empleados, para que se incluya dentro de la base de datos la inactivación de usuarios una vez se retiren. es posible seguir las recomendaciones de la guía propuesta en el proyecto en el punto Control de acceso, en la sección usuarios activos en desuso.

Las tareas de monitoreo de la base de datos haciendo uso directamente de las vistas de auditoria del sistema, o de herramientas graficas como Oracle Enterprise Manager con la que cuenta licencia la empresa, se deben contemplar en un plan a desarrollar periódicamente por parte de la oficina de informática, incluyéndolo dentro de los procedimientos a cumplir y asignando responsables claros en cada caso. Adicionalmente se recomienda configurar controles que envíen alertas al administrador de la base de datos y de esta forma poder actuar de manera preventiva a los riesgos informáticos que se puedan presentar.

10.4. POLÍTICAS DE RESPALDO Y RESTAURACIÓN

- **Pruebas de configuración de respaldo y restauración.** Se debe considerar realizar la tarea de configurar la base de datos en modo ARCHIVELOG, para potencializar los beneficios ofrecidos por Oracle en materia de respaldo y restauración. Se debe validar el escenario requerido para configurar la base de datos de esta forma, ya que se debe contar con el espacio en disco y configuración de optimización de la base de datos para que sea soportado. Para solventar este punto, es posible seguir las recomendaciones del Anexo B en la sección Configuración de modo ARCHIVELOG en la Base de Datos.

- **Políticas de respaldo** Se encontró que la empresa cuenta con configuraciones de mecanismos de respaldo óptimos y se ejecuta de forma normal el plan de respaldo que tienen establecido, contando con usuario específico de la base de datos con los privilegios exclusivos para ejecutar esta tarea.

  Se identifica un riesgo alto ya que la Oficina de tecnología no cuenta con continuidad de un administrador de base de datos, ya que existen interrupciones de periodos muy largos en la contratación de personal, y no se cuentan con personal de respaldo cuando no se encuentra el administrador principal. Se recomienda que dentro de las políticas de seguridad de la entidad se cuente con la aprobación y apoyo de la alta gerencia para solventar estos aspectos que pueden colocar en un alto riesgo los datos de la corporación.

10.5 A NIVEL DE SERVIDOR DE BASE DE DATOS

Teniendo en cuenta las vulnerabilidades encontradas en el escaneo a la seguridad del servidor de base de datos, se realizan las siguientes recomendaciones para contrarrestarlas, basadas en las soluciones sugeridas en los informes arrojados por la herramienta de escaneo Nessus utilizada para esta tarea:

- Es necesario mantener actualizado los parches que va publicando Oracle, porque traen actualizaciones con soluciones a nuevas vulnerabilidades que se van encontrando, y que en la versión original del producto no se tuvieron contempladas. Cuando se trabaja con productos Oracle es necesario contar con el soporte en estas bases de datos, para contar con soluciones rápidas y eficaces a posibles ataques por vulnerabilidades de la Base de datos.
- Se debe seguir con las recomendaciones de seguridad recomendadas y publicadas por la empresa Oracle, que se detallan por sistema operativo donde se instale y por versión de aplicación que se instale.

De igual forma se debe contar con mecanismos de seguridad en la configuración de la base de datos como:

- Contar con mecanismos robustos de autenticación de usuarios.
- Se debe contar con mecanismos de protección a nivel de sistema operativo de los archivos críticos de conexión a la base de datos que cuentan con información de acceso al servidor e información de la instancia de la base de datos, ya que los tipos de ataques que se pueden producir con vulnerabilidades a este nivel, pueden afectar por completo la base de datos en su confidencialidad, integridad y disponibilidad de los datos, y por lo general no requieren de una autenticación a la base de datos.
- Se debe restringir los permisos para acceder a archivos físicos de la base de datos y que son críticos para su funcionamiento, teniendo en cuenta el sistema operativo donde este instalada la Base de datos.
- Limitar acceso con privilegios SYSDBA, ya que este rol tiene acceso a información crítica de la Base de datos, como es el diccionario de datos de Oracle.
- Se debe tener en cuenta los controles de seguridad a nivel de Sistema Operativo, que permiten mitigar riesgos por accesos no permitidos, con la configuración optima de reglas en el Firewall; de igual forma se debe validar los privilegios que pueda tener asignado el usuario de administrador del Sistema Operativo en las bases de datos y restringir a los estrictamente necesarios.
- Una administración óptima de usuarios, donde se asignen únicamente los privilegios necesarios y se eliminen los que no son usados o están asignados a Usuarios creados por defecto en instalaciones por defecto.
- Configuración de opciones de auditoria y establecer acciones de monitoreo de la base de datos para identificar a tiempo actividades sospechosas que coloquen en riesgo la información.

11. CONCLUSIONES

Con el desarrollo del presente trabajo se logró la elaboración de una guía de configuración de seguridad de una base de datos Oracle 11g, utilizando las opciones y herramientas ofrecidas por la base de datos Oracle instalada, que dan respuesta a estándares y buenas prácticas de seguridad en cuanto a control de acceso, protección de datos, auditoria y monitoreo, mecanismos de respaldo y restauración enmarcados en el cumplimiento de las regulaciones vigentes

De esta forma la guía desarrollada en este proyecto servirá como referencia a los administradores de las bases de datos Oracle 11g en un entorno empresarial, para optimizar la configuración de seguridad de la base de dato, minimizando el riesgo generado por fallas en la configuración y potencializando los beneficios de las opciones ofrecidas por Oracle para la seguridad de los datos.

A partir de los resultados obtenidos en las pruebas de configuración de seguridad de la base de datos, planteados en la guía propuesta en este proyecto, y aplicadas a la Corporación Autónoma Regional del Valle del Cauca, se concluyó que la base de datos cuenta con varios aspectos de seguridad débiles, identificando los siguientes casos relevantes:

- Usuarios y contraseñas creados por defecto en el proceso de instalación de la base de datos, que aún se encuentran activos.
- Se identificaron más de dos usuarios con privilegios de administrador.
- Se encontraron roles de la base de datos que tienen asignados privilegios para ejecutar comandos DDL para crear, borrar o alterar tablas, otorgados a un alto número de usuarios.
- Se identificó un elevado número de usuarios que pueden ejecutar comandos DML sobre objetos del sistema, objetos de propietario SYS.
- Se encontró que la base de datos no cuenta con políticas robustas de construcción de contraseña, debido a la ausencia de una función asignada en los parámetros del profile de usuarios para que ejecute esta tarea.
- Se identifica la usencia de políticas unificadas para el encriptamiento de objetos pl/sql existentes en la base de datos. No todos los objetos paquetes o procedimientos de la base de datos, considerados como críticos por el core del negocio, se encuentran protegidos con mecanismos de encriptación.
- La base de datos no se encuentra configurada en modo archivelog, lo que limita el potencial de seguridad ofrecido por Oracle para el uso de un mecanismo de restauración más confiable y sin riesgo de pérdida de información.
-
- Como resultado del escaneo de vulnerabilidades del servidor de la base de datos, se encontró que no cuenta con todos los parches de Oracle

instalados, identificándose principalmente una vulnerabilidad en el control de tráfico entre servidor-cliente, según el plugin de la herramienta Nessus analizado: 69552 Oracle TNS Listener Remote Poisoning.

Para mejorar estas falencias de seguridad encontradas se requiere llevar a cabo una serie de acciones, basadas en la documentación de Oracle, que ofrece opciones para cerrar estas brechas de seguridad, y en el caso particular de la empresa se desarrollaron una serie de recomendaciones que fueron plasmadas en el capítulo diez de este documento, denominado 'Resultados Y Discusión', donde adicionalmente se sugieren las tareas a seguir basados en la guía de seguridad propuesta en el proyecto.

12. DIVULGACIÓN

Para el presente proyecto de grado se autoriza a la Universidad Nacional Abierta y a Distancia con el fin de que pueda disponer del presente documento para su divulgación en los diferentes formatos que considere necesarios.

El producto final de la ejecución del proyecto, será socializado con la jefatura de la oficina de Tecnologías de la Información (OTI) de la Corporación Autónoma Regional del Valle del Cauca CVC, a la cual se le entregara copia de la Guía para la administración de seguridad en Base de datos en entornos Oracle 11g.

Adicionalmente se entrega un informe ejecutivo a la jefatura de la oficina de Tecnologías de la Información de la CVC, con los resultados de las pruebas realizadas en el entorno de pruebas de la Base de Datos, socializando las vulnerabilidades encontradas y las recomendaciones que pueden ser aplicadas; Las cuáles están documentadas en la guía para la administración de seguridad en base de datos entregada.

13. BIBLIOGRAFIA

ORACLE, Documentación de Oracle Database. [en línea]. http://www.oracle.com/technetwork/es/database/enterprise-edition/documentation/database-091505-esa.html#11g.

ALONSO CEBRIÁN, José María, et al. Ataques a BB. DD., SQL injection. Seguridad en Bases de Datos, Módulo 3. Universitat Oberta de Catalunya (UOC),

ALONSO CEBRIÁN, José María, et al. Ataques a aplicaciones web. Seguridad en Bases de Datos, Módulo 2. Universitat Oberta de Catalunya (UOC),

ALONSO CEBRIÁN, José María, et al. Auditoría y desarrollo seguro. Seguridad en Bases de Datos, Módulo 4. Universitat Oberta de Catalunya (UOC),

DÍAZ SÁEZ, Vicente, Introducción a la Seguridad de Base de Datos. Seguridad en Bases de Datos, Módulo 1. Universitat Oberta de Catalunya (UOC).

SÁNCHEZ SARANGO, Ángel Fabricio. Estudio de características semánticas sobre ORACLE 11g, UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA . Escuela de Ciencias de la Computación, ciudad de Loja – Ecuador 2010

FERNÁNDEZ CUÉLLAR, Javier. Calidad y seguridad a nivel de filas en BBDD ORACLE. Universidad Carlos III de Madrid. Madrid – España 2009.

GARCÍA BASTANCHURI, David. Auditoría y control en entornos bajo ORACLE 11g. Universidad Carlos III de Madrid. Madrid – España 2013.

ISO, ISO 27000, [en línea]. http://www.iso27000.es/iso27000.html.

ISO, ISO 27001 en Español, [en línea]. http://www.iso27000.es/glosario.html

ALEXANDER SERVAT, Alberto Análisis De Riesgo Y El Sistema De Gestión De Seguridad De Información: El Enfoque ISO 270001:2005 [en línea]. < http://www.iso27000.es/download/Analisis_del_Riesgo_y_el_ISO_27001_2005>

COLOMBIA, MINISTERIO DE TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES, Ley 1273 (5 de enero de 2009). Por medio del cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado denominado “De la Protección de la información y de los datos”. [en línea]. < http://www.mintic.gov.co/portal/604/articles-3705_documento.pdf>

SQLMAP, Automatic SQL injection and database takeover tool. [en línea]. http://sqlmap.org/.

W3af, w3af's documentation [en línea]. http://docs.w3af.org/en/latest/

QUEST, Join Toad World, [en línea]. https://www.quest.com/toad/

ACUNETIX, Scan your websites, [en línea]. https://www.acunetix.com/

ISACA, COBIT 5 Spanis,. [en línea]. http://www.isaca.org/spanish/Pages/default.aspx)

HHS, Office for Civil Rights. Health Information Privacy, [en línea]. http://www.hhs.gov/hipaa/index.html

NESSUS, Tenable network security, [en línea]. https://www.tenable.com/products/nessus-vulnerability-scanner

BIS, Sobre BSI. [en línea]. http://www.bsigroup.com/es-ES/Sobre-BSI/

EcuRed, Prueba de penetración. [en línea].https://www.ecured.cu/Prueba_de_penetraci%C3%B3n

14. ANEXOS

ANEXO A - Resumen Analítico RAE

Título de Documento.	DISEÑO DE UNA GUÍA PARA LA ADMINISTRACIÓN DE LA SEGURIDAD DE BASES DE DATOS EN UN ENTORNO DE ORACLE 11G, APLICADA A LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC) EN LA CIUDAD DE CALI.
Autores	LEAL JOYA Margarita HERRERA ANGOLA Gustavo Adolfo
Palabras Claves	Base de Datos, Oracle, Buenas practicas, Vulnerabilidades
Descripción: Diseño una guía de administración de seguridad en la bases de datos Oracle 11g, que permita al administrador la implementación de controles y configuración apropiada de la base de datos, basándose en buenas prácticas y estándares de seguridad que den cumplimiento a regulaciones y normativas vigentes, aplicado en un entorno empresarial.	
Fuentes Bibliográficas	• ORACLE, Documentación de Oracle Database. [en línea]. <http://www.oracle.com/technetwork/es/database/enterprise-edition/documentation/database-091505-esa.html#11g.> • ALONSO CEBRIÁN, José María, *et al*. Ataques a BB. DD., SQL injection. Seguridad en Bases de Datos, Módulo 3. Universitat Oberta de Catalunya (UOC) • ALONSO CEBRIÁN, José María, *et al*. Ataques a aplicaciones web. Seguridad en Bases de Datos, Módulo 2. Universitat Oberta de Catalunya (UOC) • DÍAZ SÁEZ, Vicente, et al. Auditoría y desarrollo seguro. Seguridad en Bases de Datos, Módulo 4. Universitat Oberta de Catalunya (UOC). • DÍAZ SÁEZ, Vicente, Introducción a la Seguridad de Base de Datos. Seguridad en Bases de Datos, Módulo 1. Universitat Oberta de Catalunya (UOC). • SÁNCHEZ SARANGO, Ángel Fabricio. Estudio de características semánticas sobre ORACLE 11g, UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA . Escuela de Ciencias de la Computación, ciudad de Loja –

Ecuador 2010
- FERNÁNDEZ CUÉLLAR, Javier. Calidad y seguridad a nivel de filas en BBDD ORACLE. Universidad Carlos III de Madrid. Madrid – España 2009.
- GARCÍA BASTANCHURI, David. Auditoría y control en entornos bajo ORACLE 11g. Universidad Carlos III de Madrid. Madrid – España 2013.
- ISO, ISO 27000,[en línea]. http://www.iso27000.es/iso27000.html.
- ALEXANDER SERVAT**,** Alberto Análisis De Riesgo Y El Sistema De Gestión De Seguridad De Información: El Enfoque ISO 270001:2005 [en línea]. < http://www.iso27000.es/download/Analisis_del_Riesgo_y_el_ISO_27001_2005>
- MINTIC DE COLOMBIA, Ley 1273. [en línea]. < http://www.mintic.gov.co/portal/604/articles-3705_documento.pdf>
- SQLMAP, Automatic SQL injection and database takeover tool.[en línea]. http://sqlmap.org/.
- W3af, Documentation. [en línea]. <http://w3af.org/howtos>

Contenido

La información más crítica de una organización reposa en las bases de datos, y permiten la toma de decisiones eficaces y normal desarrollo de los procesos de la empresa.

En la actualidad las bases de datos son cada vez más susceptibles a ataques informáticos originados por agentes externos e internos, generalmente causados por una configuración de seguridad débil, ò por la falta de monitoreo a actividades inapropiadas realizadas en la base de datos.

Por esta razón, en el presente proyecto se plantea una solución a esta problemática específicamente para las bases de datos implementadas en Oracle 11g, en la cual se diseñó una guía de configuración de seguridad que tiene en cuenta los estándares y mejores prácticas de seguridad de la información.

La aplicación de esta guía facilitará y orientará el proceso de implementación de técnicas y mecanismos de seguridad en las bases de datos Oracle, de tal forma que garanticen la disponibilidad, integridad y confidencialidad de la información; permitiendo al administrador de la base de datos enfocar sus esfuerzos en las funciones que le conciernen de una forma más eficiente y eficaz.

Esta guía se pudo aplicar en un entorno empresarial, gracias al apoyo de las directivas de la Oficina de Tecnologías de la información de la Corporación Autónoma Regional Del Valle Del Cauca que nos dieron el aval para ejecutar las pruebas.

Objetivo General

Diseñar una guía de administración de seguridad en la bases de datos Oracle 11g, que permita al administrador la implementación de controles y configuración apropiada de la base de datos, basándose en buenas prácticas y estándares de seguridad que den cumplimiento a regulaciones y normativas vigentes.

Objetivos Específicos

- Realizar un análisis de estándares y buenas prácticas aplicadas a la seguridad en Base de datos para determinar las principales características de seguridad propias de la BD Oracle, que dan cumplimiento a regulaciones y normativas de seguridad.

- Determinar la configuración y mecanismos de respaldo de una Base de datos Oracle 11g, basada en buenas prácticas y estándares de seguridad, que den cumplimiento a normativas de seguridad de: control de acceso, auditoria, protección de información por mecanismos de encriptación, mecanismos de respaldo y restauración seguros.

- Diseñar una guía de administración de seguridad en la bases de datos Oracle 11g, basándose en buenas prácticas y estándares de seguridad que den cumplimiento a regulaciones y normativas vigentes.

- Realizar pruebas que permitan identificar las vulnerabilidades de las bases de datos en los sistemas de información de la Corporación Autónoma Regional del Valle del Cauca (CVC), con el fin de proponer posibles controles que apliquen y den cumplimiento a normativas de seguridad de las bases de datos en cuanto a control de acceso, auditoria, protección de información por mecanismos de encriptación, mecanismos de respaldo y restauración seguros.

Resumen de lo desarrollado en el proyecto

En el presente proyecto se realizó un análisis de las buenas prácticas y estándares de seguridad con las que cuenta la herramienta Oracle en su versión

11g, identificando los estándares y normativas de seguridad a las que da cumplimiento. Se planteó una guía de configuración para la optimización de la seguridad en la base de datos, utilizando las opciones que ofrece Oracle y que están acorde a los estándares y buenas prácticas de seguridad de la información y que dan respuesta a normativas vigentes de seguridad de los datos a nivel de control de acceso, auditoria y monitoreo, mecanismo de respaldo y restauración y protección de los datos. En el proyecto se llevaron a cabo pruebas de seguridad a una base de datos empresarial, de tal forma que se pudo identificar vulnerabilidades y sugerir controles de seguridad, según las alternativas propuestas en la guía.

Metodología

En presente proyecto se usa una investigación es cuantitativa, ya que pretende hacer la medición de las vulnerabilidades, amenazas y riesgos en cuanto a las características de la información de confidencialidad, integridad y disponibilidad de la información como resultado de la configuración y administración de una base de datos.

El tipo de investigación tiene componente teórico, ya que inicialmente se requiere realizar un estudio de las vulnerabilidades, amenazas y riesgos con las que cuentan actualmente las Bases de Datos en entornos Oracle 11g. Para posteriormente aplicar pruebas a la base de datos física con la que se va a desarrollar el proyecto y de esta forma validar el estado de vulnerabilidad de la Base de datos. Por último se elabora el manual para la configuración óptima de seguridad de la Base datos Oracle 11g.

La propuesta de investigación es aplicada, ya que busca resolver problemas que se dan en la práctica de la administración y configuración de una base de datos, concretamente en entorno Oracle 11g.

La investigación es de tipo explicativa, porque intenta exponer la relación entre las vulnerabilidades existentes y los ataques que pueden presentarse en una base de datos a partir de la configuración de seguridad que se le haya efectuado.

Es descriptiva porque se va a realizar una serie de mediciones de las variables relacionadas con la seguridad de la información, para poder describir cómo se va a llevar a cabo la configuración de las bases de datos, aplicando mecanismos y estándares de seguridad de la información. Se requiere de un conocimiento previo de conceptos de seguridad de la información enfocada a base de datos en entornos Oracle 11g.

Conclusiones

Con el desarrollo del presente trabajo se logró la elaboración de una guía de configuración de seguridad de una base de datos Oracle 11g, utilizando las opciones y herramientas ofrecidas por la base de datos Oracle instalada, que dan respuesta a estándares y buenas prácticas de seguridad en cuanto a control de acceso, protección de datos, auditoria y monitoreo, mecanismos de respaldo y restauración enmarcados en el cumplimiento de las regulaciones vigentes

De acuerdo a los resultado de la práctica realizada en el desarrollo del proyecto, se pudo evidenciar que la aplicación de la guía de seguridad propuesta para base de datos Oracle 11g, es útil para detectar y aplicar controles de seguridad necesarios para dar respuesta a estándares de seguridad. Ya que las falencias encontradas fue posible recomendar subsanarlas con las recomendaciones dadas en cada apartado de la guía.

De esta forma la guía desarrollada en este proyecto servirá como referencia a los administradores de las bases de datos Oracle 11g en un entorno empresarial, para optimizar la configuración de seguridad de la base de dato, minimizando el riesgo generado por fallas en la configuración y potencializando los beneficios de las opciones ofrecidas por Oracle para la seguridad de los datos.

Recomendaciones

Una vez se cuenta con los resultados de las pruebas realizadas a la base de datos empresarial, se llevan a cabo las siguientes recomendaciones de seguridad a la base de datos a nivel de:

- Control de acceso
  - Se debe validar el uso y privilegios que tienen asignados los usuarios que fueron creados por defecto en la instalación y que se encuentran aún activos. Considerar bloquearlos, o cambiarle la contraseña y privilegios asignados.
  - Limitar acceso con privilegios de administrador asignado a más de un usuario, se recomienda la desagregación de funciones de administración.
  - Validar los privilegios elevados asignados a roles que son asignados a un número alto de usuarios. Se debe revocar estos permisos que en la actualidad permiten ejecutar sentencias sobre objetos del sistema.
  - Configurar políticas de contraseñas robustas, a partir de la asignación de una función en la configuración del profile asignado en la creación de usuarios. Ya sea la función que por defecto trae Oracle para ejecutar o personalizarla.

- Protección de datos
  - Se debe contar con políticas de seguridad para los objetos de la base de dato (paquetes, procedimientos o funciones) que manejen procesos críticos de la empresa, de tal forma que se protejan con mecanismos de encriptación. La identificación de estos objetos, se debe hacer en conjunto los proveedores de los aplicativos corporativos que listen los objetos y ejecutando consultas de auditoria para identificarlos.
- Controles de auditoria
  - Se recomienda configurar controles que envíen alertas al administrador de la base de datos y de esta forma poder actuar de manera preventiva a los riesgos informáticos que se puedan presentar.
- Políticas de respaldo y restauración
  - Se debe considerar realizar la tarea de configurar la base de datos en modo ARCHIVELOG, para potencializar los beneficios ofrecidos por Oracle en materia de respaldo y restauración.
- A nivel de servidor de base de datos
  - Es necesario mantener actualizado los parches que va publicando la Oracle, ya que traen actualizaciones con soluciones a nuevas vulnerabilidades de seguridad que se van encontrando.

- Se recomienda contar con servicio de soporte de Oracle, para contar con soluciones rápidas y eficaces a posibles ataques por vulnerabilidades de la Base de dato.

ANEXO B - GUÍA DE ADMINISTRACIÓN DE SEGURIDAD EN LAS BASES DE DATOS ORACLE 11G

Tabla de Contenido

pág.

LISTA DE FIGURAS

pág.

GUÍA DE ADMINISTRACIÓN DE SEGURIDAD EN LA BASES DE DATOS ORACLE 11G

1. CONTROL DE ACCESO

1.1 CONSULTA DE PARÁMETROS DE CONFIGURACIÓN DE LA BASE DE DATOS

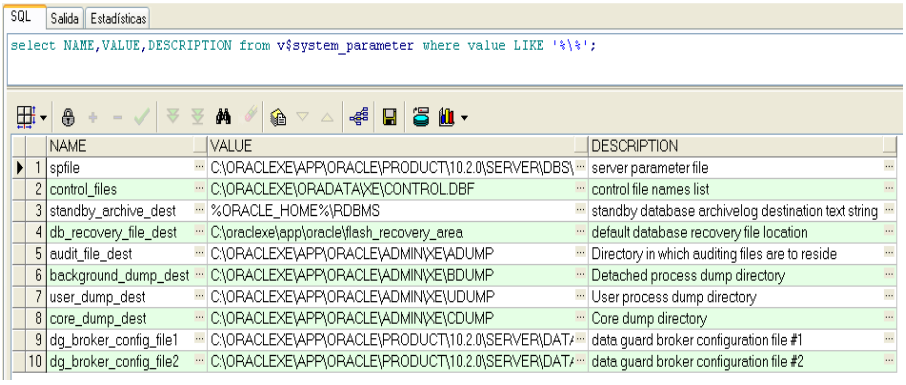
Según la documentación oficial de Oracle[67], se puede visualizar los parámetros de configuración asignados a la base de datos, a partir de la consulta “SELECT * FROM V$SYSTEM_PARAMETER;”, Se puede visualizar los parámetros de configuración asignados a la base de datos.

Para identificar las rutas de almacenamiento de los archivos propios de Oracle se usa la siguiente consulta sql:

*select * from v$system_parameter where value LIKE '%\%';*

Estas ubicaciones deben estar protegidas principalmente a nivel de sistema operativo y/o red, para evitar la degradación o fallos a nivel de base de datos como consecuencia de manipulación de archivos o directorios.

Figura 47. Consulta de configuración de la base de datos



	NAME	VALUE	DESCRIPTION
1	spfile	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DBS\	server parameter file
2	control_files	C:\ORACLEXE\ORADATA\XE\CONTROL.DBF	control file names list
3	standby_archive_dest	%ORACLE_HOME%\RDBMS	standby database archivelog destination text string
4	db_recovery_file_dest	C:\oraclexe\app\oracle\flash_recovery_area	default database recovery file location
5	audit_file_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\ADUMP	Directory in which auditing files are to reside
6	background_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\BDUMP	Detached process dump directory
7	user_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\UDUMP	User process dump directory
8	core_dump_dest	C:\ORACLEXE\APP\ORACLE\ADMIN\XE\CDUMP	Core dump directory
9	dg_broker_config_file1	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DAT/	data guard broker configuration file #1
10	dg_broker_config_file2	C:\ORACLEXE\APP\ORACLE\PRODUCT\10.2.0\SERVER\DAT/	data guard broker configuration file #2

Fuente: Propia

[67] (ORACLE CORPORATION. Oracle Help Center, Database Reference: V$SYSTEM_PARAMETER. [En línea]. 2016 Disponible en: https://docs.oracle.com/cd/E11882_01/server.112/e40402/dynviews_3097.htm#REFRN30275)

1.2. REVISIÓN DE USUARIOS POR DEFECTO

Durante la instalación de una base de datos Oracle, se crean usuarios por defecto, con privilegios y contraseñas que son de fácil acceso, debido a que se encuentran en la literatura de instalaciones de Oracle. Para evitar que la base de datos sea accedida por personas no autorizadas o ajenas a la empresa, se requiere llevar a cabo tareas para cambiar contraseñas, eliminar privilegios o bloquear los usuarios que son creados por defecto en la instalación.

Para identificar los usuarios que fueron creados en el proceso de instalación de la base de datos y que aun cuentan con la contraseña que se asigna por defecto, se consulta la vista '*DBA_USERS_WITH_DEFPWD*'. Según la documentación oficial de Oracle[68] para identificar de forma rápida estos usuarios se ejecuta la siguiente consulta sql:

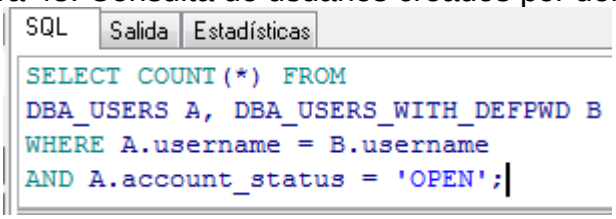
*"SELECT * FROM DBA_USERS_WITH_DEFPWD;"*

De los usuarios que se encuentren, ejecutando la anterior consulta es necesario conocer, cuántos de ellos se encuentran aún activos en la base de datos, y que puedan ser utilizados por una persona no autorizada. Para esto es necesario hacer uso de la vista DBA_USERS, según la documentación oficial de Oracle[69], esta vista cuenta con la información de las cuentas de usuarios de la base de datos y el estado de la cuentas.

Haciendo uso de la vista *DBA_USERS y la tabla DBA_USERS_WITH_DEFPWD , es posible* conocer de los usuarios creados por defecto, cuantos se encuentran activos en la base de datos, La sentencia sql a ejecutar es:

SELECT COUNT(*) FROM DBA_USERS A, DBA_USERS_WITH_DEFPWD B WHERE A.username = B.username AND A.account_status = 'OPEN';

Figura 48. Consulta de usuarios creados por defecto



[68] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS_WITH_DEFPWD. [En línea]. 2016 Disponible en:https://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5074.htm)
[69] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS [En línea]. 2016, Dispoible en: https://docs.oracle.com/cd/E18283_01/server.112/e17110/statviews_5081.htm)

Fuente: Propia

Los usuarios que se encuentren activos con contraseñas por defecto, es necesario, cambiar la contraseña o bloquearlos según se requiera. Para bloquearlos la sentencia sql a ejecutar es:

ALTER USER [NombreUsuario] ACCOUNT LOCK;

Para llevar a cabo un cambio de contraseña, la sentencia sql a ejecutar es:

alter user [Nombre_usuario] identified by [Nueva_contraseña];

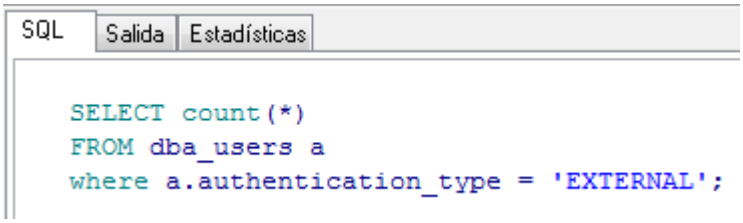
1.3. PRIVILEGIOS DE USUARIO DE SISTEMA OPERATIVO CON PERMISOS ELEVADOS EN LA BASE DE DATOS

Los usuarios con autenticación de sistema operativo, según la documentación oficial de Oracle[70], se pueden identificar por ser creados normalmente con la característica de tipo de autenticación "EXTERNAL", que les permite acceder a la base de datos sin ingresar contraseña.

Por defecto en la instalación se crea el usuario de autenticación de sistema operativo con privilegios de administrador. Debido a esto el tipo de usuario no requiere autenticación para acceder a la base de datos, se debe validar sus privilegios en la base de datos, ya que normalmente el administrador de la base de datos no es el mismo administrador del sistema operativo, esto para garantizar un nivel mayor de seguridad, es recomendable que el usuario del sistema operativo solo cuente con los privilegios de conexión. Para validar usuarios con tipo de autenticación se ejecuta la siguiente sentencia sql:

"SELECT count(*) FROM dba_user A WHERE a.authentication_type = 'EXTERNAL'".

Figura 49. Consulta de usuarios autenticación EXTERNAL



Fuente: Propia

[70] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_USERS [En línea]. 2016, Dispoible en: https://docs.oracle.com/cd/E18283_01/server.112/e17110/statviews_5081.htm)

El usuario de sistema operativo son creados con el prefijo ‘OPS$’, lo cual también es posible consultarlo dentro de la misma vista de DBA_USER, con la siguiente consulta sql:

*SELECT * FROM dba_users a where a.username like '%OPS$%'*

Según la documentación oficial de Oracle,[71] para validar si el usuario de sistema operativo cuenta con privilegios de administrador, es posible consultarlo por la vista dba_role_privs , ejecutando la siguiente sentencia sql:

```
select count(*)
from dba_users a, dba_role_privs b
where a.username = b.GRANTEE
and a.username like '%OPS$%'
and b.GRANTED_ROLE = 'DBA';
```

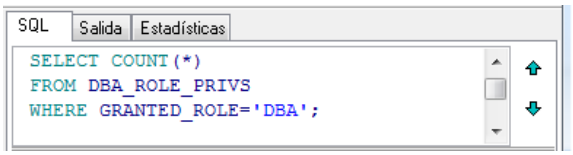
Si se encontrara que el usuario de sistema operativo cuenta con privilegios de administrador, es necesario retirarlos ya que es un usuario con autenticación external, lo que implica que accede a la base de datos sin necesidad de escribir la contraseña, lo que implicaría un riesgo que un atacante solo conociendo el nombre del usuario de sistema operativo ingrese a la base de datos con privilegios de administrador sin que se le solicite contraseña.

1.4. VALIDACIÓN DE USUARIOS CON PERMISOS DE ADMINISTRADOR

Para conocer los usuarios con rol de administrador, según la documentación oficial Oracle[72], se hace una consulta sobre la vista de Oracle DBA_ROLE_PRIVS, que muestra los privilegios asignados a usuarios y roles de la base de datos. La consulta a la vista se hace referenciando el rol DBA, para conocer los usuarios con permisos elevados. La sentencia sql a ejecutar:

```
SELECT COUNT(*) FROM DBA_ROLE_PRIVS WHERE GRANTED_ROLE='DBA';
```

Figura 50. Consulta de usuarios con permiso DBA



Fuente: Propia

[71] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_ROLE_PRIVS. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_4206.htm)
[72] Ibíd.

Según la documentación oficial de Oracle[73], para quitar un privilegio a un usuario se ejecuta la sentencia:

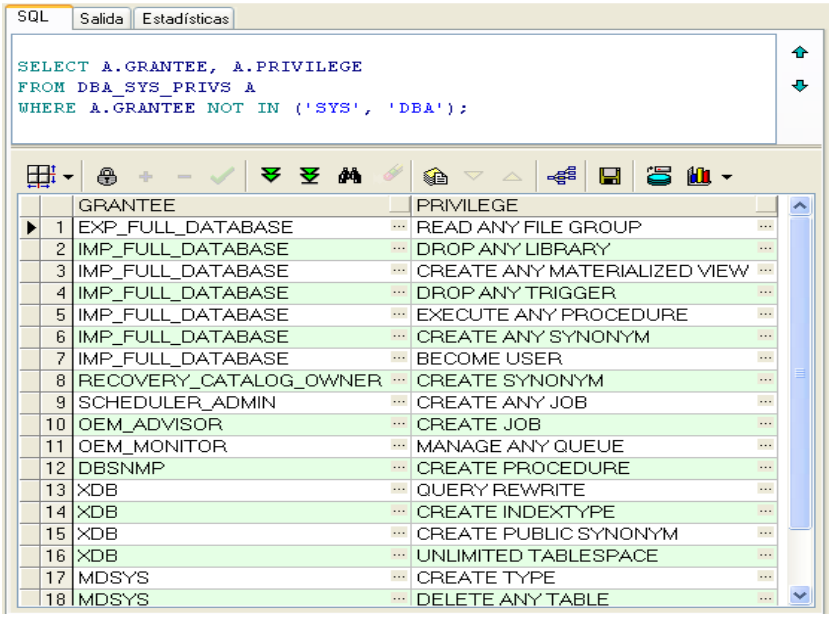
REVOKE role FROM {user, | role, |PUBLIC}

Es recomendable realizar la ejecución de esta sentencia periódicamente ya que permite buscar inconsistencias en el número de usuarios con privilegios DBA, que posiblemente hayan sido creados por un atacante.

Realizar tareas de validación de privilegios de los usuarios periódicamente, permite controlar el acceso no autorizado a la base de datos y tomar las acciones pertinentes para garantizar que los permisos de los usuarios sobre los objetos sean los mínimos requeridos.

Otro objeto de la base de datos que permiten consultar información relacionada a los roles del sistema, según la documentación oficial de Oracle[74], es la vista *DBA_SYS_PRIVS* . En la siguiente figura se muestra un ejemplo de una consulta a esta vista:

Figura 51. Consulta de usuarios con permiso DBA por la vista DBA_SYS_PRIVS



	GRANTEE	PRIVILEGE
1	EXP_FULL_DATABASE	READ ANY FILE GROUP
2	IMP_FULL_DATABASE	DROP ANY LIBRARY
3	IMP_FULL_DATABASE	CREATE ANY MATERIALIZED VIEW
4	IMP_FULL_DATABASE	DROP ANY TRIGGER
5	IMP_FULL_DATABASE	EXECUTE ANY PROCEDURE
6	IMP_FULL_DATABASE	CREATE ANY SYNONYM
7	IMP_FULL_DATABASE	BECOME USER
8	RECOVERY_CATALOG_OWNER	CREATE SYNONYM
9	SCHEDULER_ADMIN	CREATE ANY JOB
10	OEM_ADVISOR	CREATE JOB
11	OEM_MONITOR	MANAGE ANY QUEUE
12	DBSNMP	CREATE PROCEDURE
13	XDB	QUERY REWRITE
14	XDB	CREATE INDEXTYPE
15	XDB	CREATE PUBLIC SYNONYM
16	XDB	UNLIMITED TABLESPACE
17	MDSYS	CREATE TYPE
18	MDSYS	DELETE ANY TABLE

Fuente: Propia

[73] (ORACLE CORPORATION. Oracle Help Center, Database Reference: REVOKE. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_9020.htm)
[74] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_SYS_PRIVS. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5036.htm)

1.5 VALIDACIÓN DE USUARIOS CON PRIVILEGIOS PARA EJECUTAR COMANDOS DDL

Según la documentación oficial de Oracle[75], la base de datos cuenta con la vista SYSTEM_PRIVILEGE_MAP, en la que se puede consultar todos los privilegios con los que cuenta Oracle, los cuales pueden ser asignados a usuarios o roles.

En la vista SYSTEM_PRIVILEGE_MAP se encuentran los privilegios para ejecutar comandos DDL (data definition language), como como CREATE, ALTER y DROP. Según la documentación oficial de Oracle[76], este tipo de comandos permite definir estructuras de objetos de la base de datos.

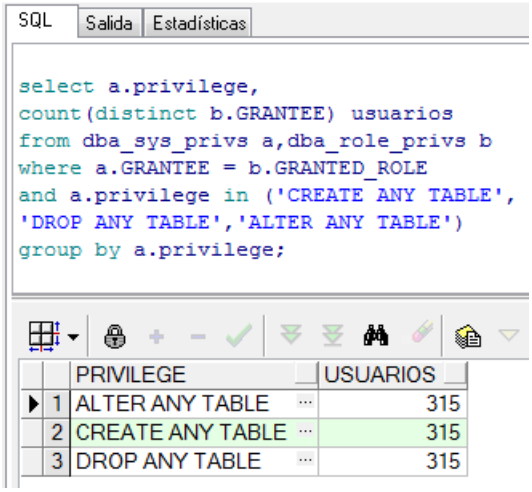
Para validar los usuarios que actualmente cuentan con permisos en el sistema para ejecutar comandos DDL como CREATE, ALTER y DROP sobre todas las tablas, se ejecuta la siguiente sentencia sql:

```
select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by a.privilege;
```

[75] (ORACLE CORPORATION. Oracle Help Center, Database Reference: SYSTEM_PRIVILEGE_MAP. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5158.htm)

[76] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: Types of SQL Statements. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_1001.htm)

Figura 52. Usuarios con permisos DDL

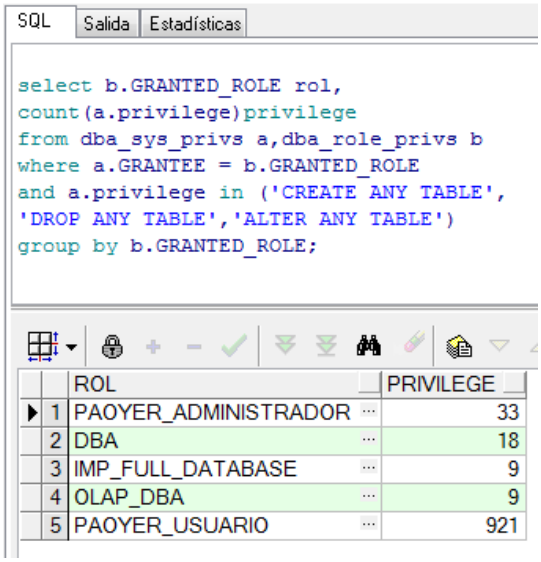


Fuente: Propia

Para consultar los roles que cuentan con permisos para crear, alterar o borrar tablas, se ejecuta la siguiente sentencia SQL:

select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('CREATE ANY TABLE',
'DROP ANY TABLE','ALTER ANY TABLE')
group by b.GRANTED_ROLE;

Figura 53. Roles con permisos DDL



Fuente: Propia

Con el resultado de estas consultas, el administrador de la base de datos, puede identificar y monitorear los usuarios que deben tener este tipo de privilegios.

1.6 VALIDACIÓN DE USUARIOS CON PRIVILEGIOS PARA EJECUTAR COMANDOS DML

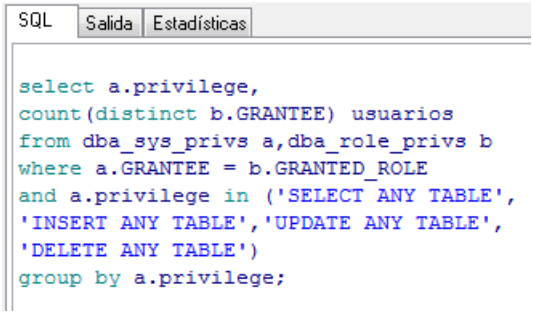
Las sentencias de lenguaje de manipulación de datos (Data Manipulation Language DML), según la documentación oficial de Oracle[77], son utilizadas para realizar tareas de consulta, inserción, modificación y eliminación de los datos contenidos en las bases de datos, por esta razón se recomienda conocer los usuarios que cuentan con los permisos para realizar este tipo de actividades sobre objetos del sistema y la ejecución de monitoreo que permita la identificación de los usuarios existentes y validación de los mismos para conocer la cantidad de usuarios con permisos en el sistema para ejecutar comandos DML (Data manipulation language), como SELECT, INSERT, UPDATE y DELETE sobre todas las tablas, se ejecuta la siguiente sentencia sql:

select a.privilege,
count(distinct b.GRANTEE) usuarios
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE

[77] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: Types of SQL Statements. [En línea]. 2016, Dispoible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_1001.htm)

and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by a.privilege;

Figura 54. Usuarios con permisos DML

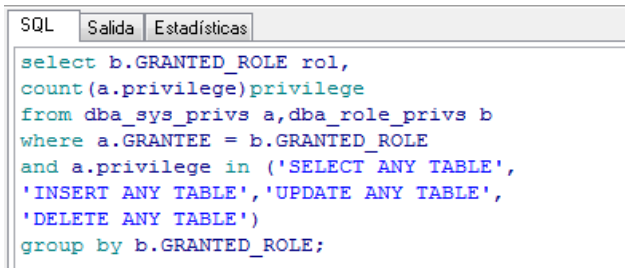


Fuente: Propia

Para consultar los roles que cuentan con permisos para seleccionar, modificar, actualizar o borrar tablas, se ejecuta la siguiente sentencia SQL:

select b.GRANTED_ROLE rol,
count(a.privilege)privilege
from dba_sys_privs a,dba_role_privs b
where a.GRANTEE = b.GRANTED_ROLE
and a.privilege in ('SELECT ANY TABLE',
'INSERT ANY TABLE','UPDATE ANY TABLE',
'DELETE ANY TABLE')
group by b.GRANTED_ROLE;

Figura 55. Roles con permisos DML



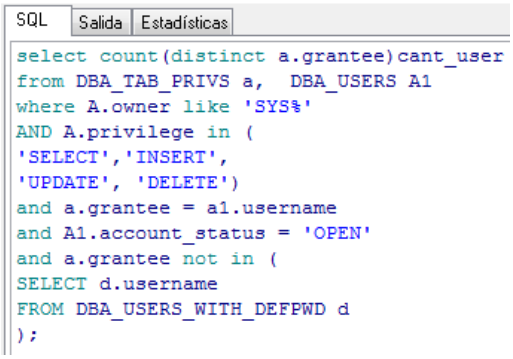
Fuente: Propia

Otra forma de conocer los usuarios con permisos para ejecutar comandos DML sobre objetos del sistema es haciendo referencia la vista DBA_TAB_PRIVS, según

la documentación oficial de Oracle[78] esta vista describe todos los privilegios asignados a los objetos en la base de datos referenciando al propietario del objeto. se ejecuta la siguiente sentencia SQL:

```
select count(distinct a.grantee)cant_user
from DBA_TAB_PRIVS a,  DBA_USERS A1
where A.owner like 'SYS%'
AND A.privilege in (
'SELECT','INSERT',
'UPDATE', 'DELETE')
and a.grantee = a1.username
and A1.account_status = 'OPEN'
and a.grantee not in (
SELECT d.username
FROM DBA_USERS_WITH_DEFPWD d
);
```

Figura 56. Consulta a usuarios en la vista DBA_TAB_PRIVS



Fuente: Propia

Con el resultado de estas consultas, el administrador de la base de datos, puede identificar y monitorear los usuarios que deben tener este tipo de privilegios y tomar la acción de revocarlos o bloquearlos en el caso que sea necesario.

1.7. POLÍTICAS DE CONTRASEÑAS

Según la documentación oficial de Oracle[79], la consulta de perfiles de la base de datos y su configuración es posible conocerla por medio de la vista DBA_PROFILES, que se basa en la tabla sys.profile$. Cada perfil de la base de

[78] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_TAB_PRIVS. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_5046.htm)
[79] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: DBA_PROFILES. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28320/statviews_4175.htm)

datos define los límites de los recursos de la base de datos que se la asignaran a los usuarios que se creen, a nivel de contraseñas (password_parameters) y de límites de recursos asignados de servidor (resource_parameters). Según la documentación oficial de Oracle[80], Los principales parámetros de PASSWORD,en los perfiles son los siguientes:

- PASSWORD_LIFE_TIME: especifica el tiempo de vigencia de la contraseña.
- PASSWORD_GRACE_TIME: especifica el tiempo límite en días para la modificación de la contraseña, antes de que se bloquee la cuenta.
- PASSWORD_REUSE_TIME: tiempo en días para utilizar el mismo password
- PASSWORD_REUSE_MAX: Permite definir el número máximo de veces que es posible reusar la misma clave.
- FAILED_LOGIN_ATTEMPTS: número máximo de intentos fallidos de ingreso de la clave, antes de que se bloquee la cuenta de usuario, se recomienda tres intentos como máximo
- PASSWORD_LOCK_TIME : Especifica el número de días en la que una cuenta estará bloqueada después de cumplir el número de intentos fallidos consecutivos para acceder. Si no se especifica un valor, por defecto es un día.
- PASSWORD_VERIFY_FUNCTION: En este parámetro permite asignar una función para establecer la complejidad de la contraseña que crean los usuarios. Por ejemplo exigir una longitud mínima, que no se asigne el mismo nombre del usuario, que cuente con caracteres especiales, entre otras características que se puedan asignar a la función.

Cuando este parámetro PASSWORD_VERIFY_FUNCTION es null, no se cuenta con ninguna regla para la creación de contraseñas. Según la documentación oficial de Oracle[81], la base de datos cuenta por defecto con una función, 'verify_function_11G', para asignar en este parámetro, que es posible crearla ejecutando, con usuario sys, el archivo utlpwdmg.sql que se encontraría en: $ORACLE_HOME/rdbms/admin.

Para cambiar el valor del parámetro PASSWORD_VERIFY_FUNCTION, de tal forma que tome una función de verificación de contraseñas, según la

[80] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm)

[81] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Configuring Authentication. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/network.111/b28531/authentication.htm)

documentación oficial de Oracle[82] se utiliza *ALTER PROFILE,* ejecutando la siguiente sentencia sql:

ALTER PROFILE default LIMIT
PASSWORD_VERIFY_FUNCTION verify_function_11G;

De igual forma es posible asignar una función personalizada, construyendo su código en PL/SQL y posteriormente alterar el perfil en su parámetro *PASSWORD_VERIFY_FUNCTION, asignando la nueva función*

Un ejemplo de configuración de perfiles de password, con una función personalizada (*Validar_Pass*) para validación de contraseñas seria:

CREATE OR REPLACE PROFILE Politica_Pass LIMIT
PASSWORD_LIFE_TIME 45
PASSWORD_GRACE_TIME 7
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 1
PASSWORD_VERIFY_FUNCTION Validar_Pass;

Adicionalmente se pueden establecer controles a nivel se servidor configurando los parámetros de resource_parameters, según la documentación oficial de Oracle[83], estos parámetros están más relacionados con la configuración de optimización de la base de datos. En la siguiente tabla se detallan algunos de estos parámetros:

Tabla 8 - Descripción de parámetros resource_parameters

Campo	Descripción
SESSIONS_PER_USER	Numero límite de sesiones simultáneas de un usuario
CPU_PER_SESSION	Especifica el límite de tiempo de CPU para una sesión, expresado en centésimas de segundos.
Fuente: ORACLE CORPORATION. Oracle Help Center,Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm	

[82] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: ALTER PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_2007.htm)
[83] (ORACLE CORPORATION. Oracle Help Center, Database SQL Language Reference: CREATE PROFILE. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_6010.htm)

Tabla 9. (Continuación)

Campo	Descripción
CPU_PER_CALL	Especifica el límite de tiempo de CPU para una llamada a la base de datos (un análisis sintáctico, ejecutar, o ir a buscar), expresado en centésimas de segundos.
CONNECT_TIME	Especifica el límite de tiempo total transcurrido para una sesión
IDLE_TIME	Especifica los períodos permitidos de tiempo de inactividad continua durante una sesión, expresado en minutos.
LOGICAL_READS_PER_SESSION	Especifica el número permitido de bloques de datos en una sesión de lectura, incluyendo los bloques leídos de la memoria y el disco.
LOGICAL_READS_PER_CALL	Especifica el número permitido de bloques de datos leídos por una llamada para procesar una instrucción SQL (un análisis sintáctico, ejecutar, o ir a buscar).
PRIVATE_SGA	Especifica la cantidad de espacio privado que una sesión puede asignar en el shared pool del SGA.
COMPOSITE_LIMIT	Especifica el costo total de recursos para una sesión, expresado en unidades de servicio. Calculando las unidades de servicio como una suma ponderada de los parámetros: CPU_PER_SESSION, CONNECT_TIME, LOGICAL_READS_PER_SESSION, y PRIVATE_SGA

1.8. ACCESO RESTRINGIDO A INFORMACIÓN DE BD REMOTAS

Para permitir que usuarios accedan a objetos de una base de datos remota con privilegios restringidos, sin estar creado como usuario en esa base de datos, es aconsejable el uso de bdlink. Según la documentación oficial de Oracle [84]de esta forma se accede a una base remota con un usuario que no está creado en esa base de datos, limitando sus permisos. Para la creación del DBLlink, se debe

[84] (ORACLE CORPORATION. Oracle Help Center, Database Administrator's Guide: Database Links. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28310/ds_concepts002.htm#i1007709)

tener en cuenta que se ingresen datos de conexión de un usuario existente en la base de datos a la que se quiere crear la conexión remota.

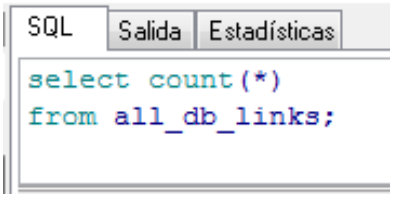
CREATE SHARED PUBLIC DATABASE LINK [NOMBRE_DB_LINK]
CONNECT TO [NOMBRE_DEL_USUARIO_DE_BASE_DATOS] IDENTIFIED BY VALUES ':1'
AUTHENTICATED BY [CLAVE_DEL_USUARIO] IDENTIFIED BY VALUES ':2'
USING '//[IP_DE_BASE_DATOS]:[PUERTO]/[NOMBRE_BASE_DATOS]';

Teniendo en cuenta que se convierte en una buena práctica para permitir acceso a una base de datos remota sin crear usuario local en esa base de datos, es aconsejable que el administrador de la base de datos, realice tareas periódicas para validar si se cuenta con esa buena práctica y a su vez validar que no se hayan creado DBLlink que no hayan sido autorizadas. Para consultar los DBLlink existentes en la base de datos se ejecuta la siguiente sentencia sql:

SELECT * FROM DBA_DB_LINKS;

Para conocer la cantidad de DBLlink, creados, se ejecuta: *select count(*) from DBA_DB_LINKS;*

Figura 57. Cantidad de DBLINK creados en la base de datos



Fuente: Propia

1.9. BUENAS PRÁCTICAS EN EL DESARROLLO DE LAS APLICACIONES

- ✓ Validar que dentro de código fuente de los programas no haya usuarios y contraseñas explicitas.
- ✓ Evitar incluir código PL dentro del código fuente, procurar hacer uso de llamados a packages de la BD y estos a su vez deben pasar por procesos de encriptación.
- ✓ Evitar incluir en código fuente nombres de directorios , tablas, ID, de la BD

- ✓ Dentro de las políticas de seguridad de la organización , se debe incluir tareas periódicas que respondan a controles preventivos a la seguridad de la Base de datos como:
  - La ejecución de tareas de pententing, para validar vulnerabilidades de las aplicaciones. Haciendo uso de herramientas como Sqlmap para pruebas de pentest a aplicaciones en ambiente WEB
  - La validación periódica de privilegios de usuario, teniendo en cuenta privilegios de administrador, al igual que se debe contar con procedimientos precisos de la revocación de permisos de usuarios retirados.

2. PROTECCIÓN DE DATOS

La confidencialidad de la información, toma una mayor relevancia en los procesos donde involucra el manejo de datos sensibles dentro de los sistemas informáticos, dada la vulnerabilidad que tiene al estar expuestos en la red. Oracle cuenta con mecanismos de seguridad a nivel de encriptación de los objetos de la base de datos. A continuación uno de estos mecanismos.

2.1. A NIVEL DE CÓDIGO PL/SQL ALMACENADA EN LA BASE DE DATOS

Dentro de las buenas prácticas de programación en pl/sql se recomienda empaquetar los objetos que son usados para ejecutar un proceso en común, o porque el conjunto e interacción de estos objetos permite ejecutar alguna funcionalidad del negocio. De esta forma se facilita la administración y monitoreo del código PL/SQL utilizado.

El código PL/SQL, que se almacena en las bases de datos, es recomendable ocultarlo, principalmente para los objetos que manejan código sensible o crítico del negocio. De esta forma se protege el código fuente de la competencia del negocio, se evita daños voluntarios o involuntarios del código, y se protege a su vez propiedad intelectual del propio programador.

Oracle cuenta con herramientas que permiten ocultar información correspondiente a código PL/SQL almacenado en la BD, ya sea funciones, procedimientos o paquetes. Según la documentación oficial de Oracle[85], cuenta con la utilidad

[85] (ORACLE CORPORATION. Oracle Help Center, Database PL/SQL Language Reference: A Wrapping PL/SQL Source Code. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/B28359_01/appdev.111/b28370/wrap.htm)

WRAP, es una de las funcionalidades con las que cuenta Oracle para llevar a cabo esta tarea, por medio de mecanismos de encriptación.

Según la empresa líder de consultoría en soporte y formación en Oracle, Burleson Colsulting[86], los pasos a seguir para encriptar objetos de la base de datos que contienen código PL/SQL son:

- Se identifica el objeto que almacena el código PL/SQL en la base de datos a encriptar, generando el archivo SQL para su creación en la base de datos. En el caso de los paquetes, se recomienda ocultar solamente el cuerpo, ya que la cabecera es preferible dejarla legible, para identificar la descripción de la funcionalidad del objeto, facilitando cualquier procedimiento de pruebas o calidad en la lógica de programación.
- Haciendo uso del ejecutable wrap, por consola, se ejecuta la siguiente instrucción:

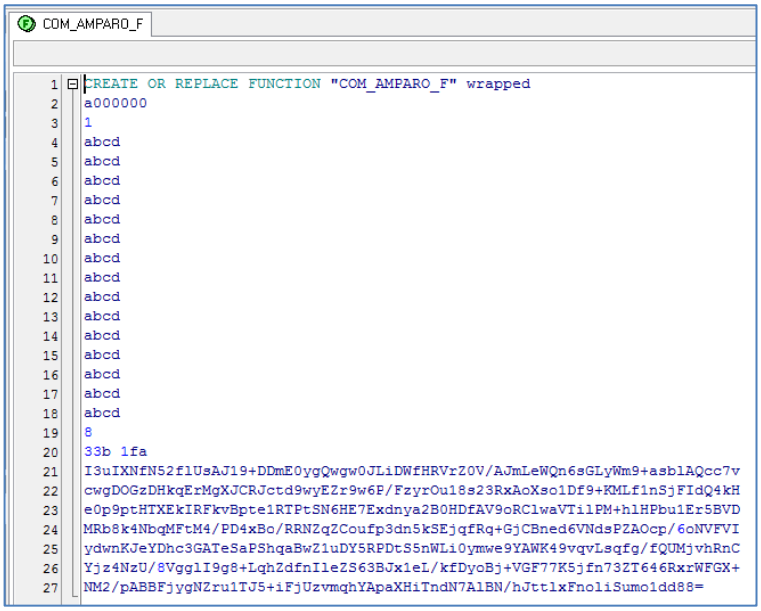
  WRAP iname=NombreArchivoObjetoPL.sql
  Lo que genera un archivo de texto con extensión .plb. De esta forma se crea el archivo NombreArchivoObjetoPL.plb que contiene el código fuente convertido.
- El código creado en el archivo *.plb, se compila en una consola de SQL*plus para crear el objeto encriptado en la BD.

  SQL> @ NombreArchivoObjetoPL.plb

  Al buscar nuevamente el objeto en la base de datos se visualiza de la siguiente forma:

[86] (Burleson Colsulting. Oracle Wrap Utility. [En línea]. 2016, Disponible en:http://www.dba-oracle.com/t_wrap_utility.htm)

Figura 58. Ejemplo de objeto de base de datos encriptado



Fuente: Propia

Es recomendable como buena práctica de programación en PL/SQL, que se maneje versionamiento del código creado, utilizando repositorios de código, ya que las modificaciones al mismo se deben hacer sobre el código original repitiendo el mismo proceso para compilar.

3. AUDITORIA Y MONITOREO.

Para garantizar la confiabilidad de la información debemos contar con técnicas o métodos que permitan realizar la trazabilidad de la misma, con el fin de identificar cambios, para esto necesario la configuración de mecanismos de auditoria en una base de datos, ya que permite realizar seguimientos al uso de la base de datos, posibilitando llevar a cabo acciones preventivas o correctivas al identificar actividades en la base de datos que puedan estar produciendo algún riesgo; Entre las acciones que se registran en la auditoria de la base de datos, cuando está activa están: auditoria de inicios de sesión, auditoria de acceso a objetos y auditoria de acciones sobre objetos de la base de datos.

3.1. ACTIVACIÓN DE AUDITORIA DE ORACLE

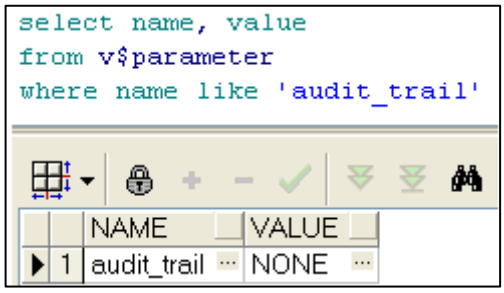
Según la documentación oficial de Oracle[87], la información de las auditorias en Oracle es almacenada en diccionario de Base de Datos dentro de la tabla

[87] (ORACLE CORPORATION. Oracle Help Center, Database Reference: AUDIT_TRAIL. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28320/initparams017.htm)

SYS.AUD$. Para que los datos de auditoria sean guardados, es necesario que se encuentre habilitada en el parámetro de configuración *'audit_trail'*. Para validar si la instancia de la Base de datos, tiene activa la auditoria, se consulta el parámetro de la base de datos :'audit_trail', ejecutando la siguiente instrucción SQL:

```
select name, value
from v$parameter
where name like 'audit_trail'
```

Figura 59. Visualizar el estado de la auditoria propia de Oracle



Fuente: Propia

Los valores que puede tomar el parámetro *'audit_trail'* son[88]:

- ✓ none: deshabilita la auditoría de la base de datos.
- ✓ db: habilita la auditoría, almacenando los datos en taba SYS.AUD$.
- ✓ os: habilita la auditoría de la base de datos, donde el Sistema Operativo se encarga de la auditoria de los sucesos auditados
- ✓ bd, extended: habilita la auditoría, almacenando los datos en SYS.AUD$, y adicionalmente se registran datos en la columna SQLBIND y SQLTEXT de la tabla SYS.AUD$.
- ✓ xml: habilita la auditoría, las actividades registradas se escriben en archivos XML del Sistema Operativo
- ✓ xml, extended habilita la auditoría, las actividades registradas se escriben en formato XML del sistema operativo, se incluyen valores de

Por defecto en la instalación, Oracle trae por defecto el valor 'none' en el parámetro *'audit_trail'.*

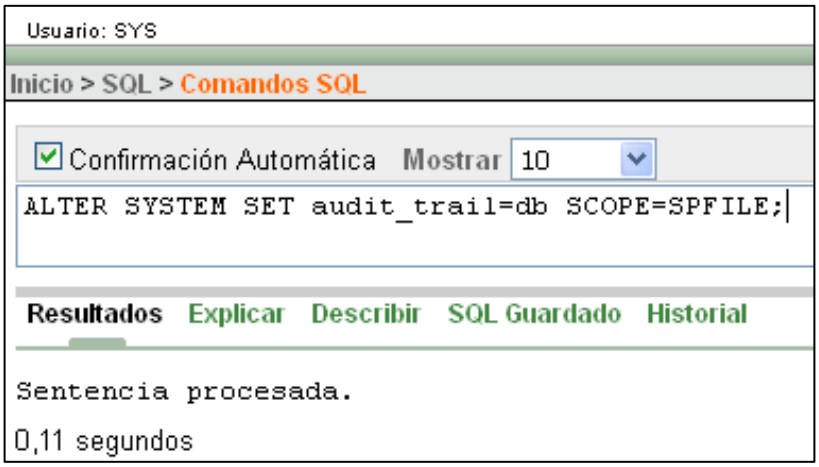
Los pasos para habilitar la auditoria de la base de datos, según la documentación oficial de Oracle, son los siguientes: [89]

[88] (ORACLE CORPORATION. Oracle Help Center, Database Reference: AUDIT_TRAIL. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28320/initparams017.htm)

- Se ejecuta La instrucción sql:

ALTER SYSTEM SET audit_trail=db SCOPE=SPFILE;

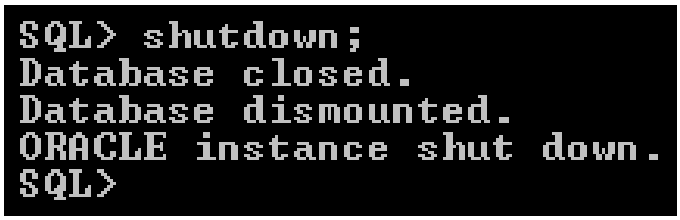
Figura 60. Habilitar auditoria Oracle



Fuente: Propia

- Reiniciar la base de datos para que tome los cambios:

Figura 61. Instrucción para bajar la Base de Datos



Fuente: Propia

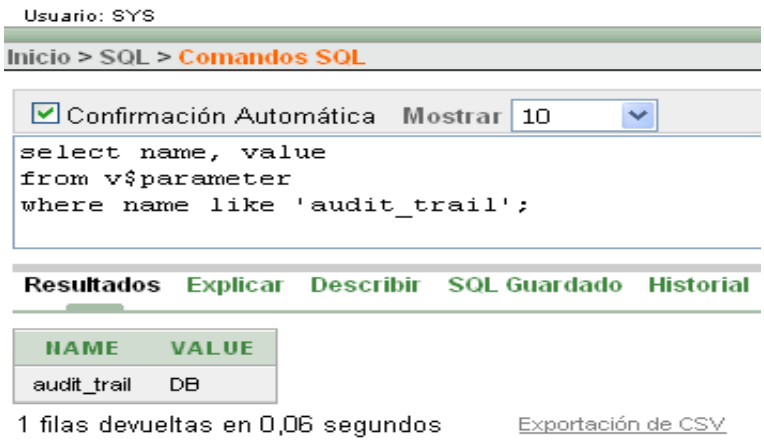
Figura 62. Instrucción para subir la Base de Datos

Fuente: Propia

[89] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Verifying Security Access with Auditing. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/network.112/e36292/auditing.htm#DBSEG006)

- Validar que haya tomado el cambio en el parámetro para habilitar la auditoria en la base de datos:

Figura 63. Validar activación de auditoria Oracle

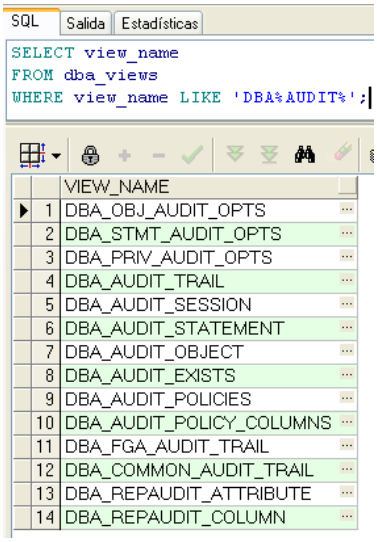


Fuente: Propia

Para detallar el contenido de la tabla $sys.aut, es posible hacerlo a través de las siguientes vistas de la base de datos:

SELECT view_name FROM dba_views WHERE view_name LIKE 'DBA%AUDIT%';

Figura 64. Consultar vistas de auditoria



Fuente: Propia

Oracle ofrece diferentes vistas de auditoria, basadas en la tabla SYS.AUD$, y FGA_LOG$, que ofrecen diferentes puntos de vista de los registros de auditoria de la base de datos, según la documentación oficial de Oracle [90]se encuentran las principales vistas de auditoria:

- DBA_AUDIT_OBJECT: Se muestra información detallada de la auditoría de objetos de la base de datos
- DBA_AUDIT_SESSION: Se muestra información detallada de la auditoría de los inicios de sesión de usuario.
- DBA_AUDIT_TRAIL: Se muestra información detallada de la auditoría estándar.
- USER_AUDIT_TRAIL: Se muestra información detallada de la auditoría estándar, del usuario actual. Todos los usuarios cuentan con esta vista.
- DBA_FGA_AUDIT_TRAIL: Se muestra información detallada de la auditoría de grano fino (FGA), obtenida de la tabla FGA_LOG$. La auditoría FGA extiende la auditoría estándar de conocer cual usuario ejecuta una determinada acción sobre un objeto, para conocer adicionalmente la auditoria a la sentencia sql que ejecuta el usuario sobre el objeto, los datos que fueron creados, borrados o actualizados por parte del usuario.

3.2. VALIDACIÓN DE USUARIOS ACTIVOS EN DESUSO:

Dentro de las tablas de auditoria de Oracle, se encuentra la de auditoria de inicio de sesión de los usuarios, según la documentación oficial de Oracle[91], se encuentra la vista DBA_AUDIT_SESSION que es una vista útil para realizar un monitoreo a los usuarios activos de una base de datos pero que no hayan tenido actividad en ella en un tiempo determinado. El tiempo de inactividad prudente, lo define el dba o usuario encargado del monitoreo de la base de datos. Una consulta para identificar estos usuarios es la siguiente:

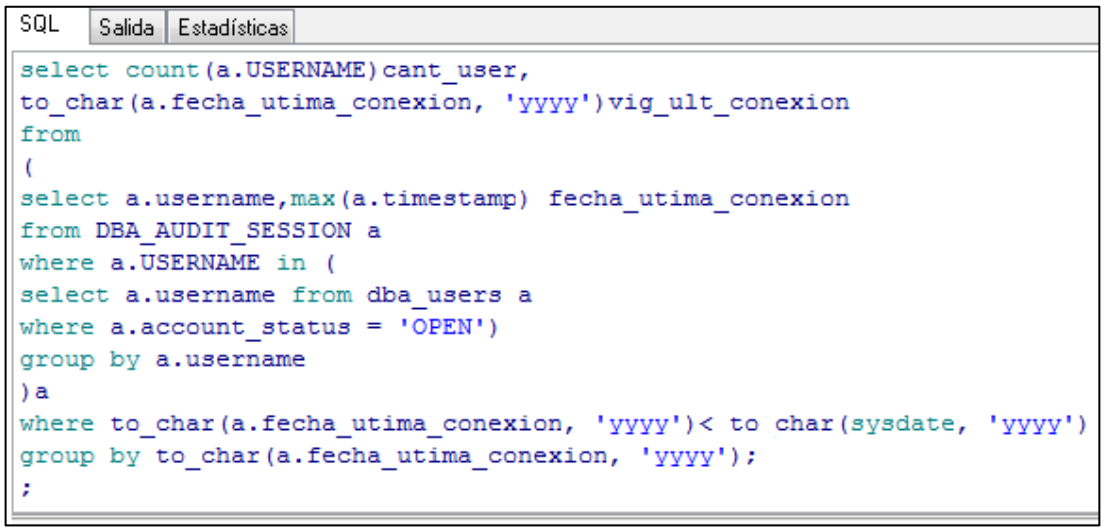
```
select count(a.USERNAME)cant_user,
to_char(a.fecha_utima_conexion, 'yyyy')vig_ult_conexion
from
(
select a.username,max(a.timestamp) fecha_utima_conexion
from DBA_AUDIT_SESSION a
where a.USERNAME in (
```

[90] (ORACLE CORPORATION. Oracle Help Center, Database Security Guide: Verifying Security Access with Auditing. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/network.112/e36292/auditing.htm#DBSEG006)

[91] (ORACLE CORPORATION. Oracle Help Center, Database Reference: DBA_AUDIT_SESSION. [En línea]. 2016, Disponible en: http://docs.oracle.com/cd/E11882_01/server.112/e40402/statviews_3079.htm#REFRN23021)

select a.username from dba_users a
where a.account_status = 'OPEN')
group by a.username
)a
where to_char(a.fecha_utima_conexion, 'yyyy')< to_char(sysdate, 'yyyy')
group by to_char(a.fecha_utima_conexion, 'yyyy');

Figura 65. Usuarios activos obsoletos en la base de datos



Fuente: Propia

Se toman todos los usuarios activos (dba_users) se comparan con los usuarios que ha realizados inicio de sesión en años anteriores al año actual (DBA_AUDIT_SESSION); Obteniendo el listado de los usuarios activos que no han realizado conexiones a la base de datos en el año actual, información que permite identificar a usuarios que deben ser cambiados a estado inactivo.

Al identificar los usuarios se debe proceder a bloquearlos, ya que pueden pertenecer a usuarios retirados de la organización o usuarios que ya no cuentan con actividades donde tengan que intervenir con información de la base de datos. Dentro de las buenas prácticas de seguridad en la organización se debe tener en cuenta incluir las tareas de bloqueo inmediato de personas retiradas de la organización y de reasignación de roles en la base de datos, cuando los usuarios son cambiados de funciones en la empresa.

4. CONFIGURACIÓN DE RESPALDO Y RESTAURACIÓN DE LA BASE DE DATOS

Para garantizar la conservación de la información y la continuidad del negocio, se requiere contar con métodos y herramientas que permitan la pronta recuperación y minimicen la pérdida de datos; Para cumplir este objetivo se establecen políticas de seguridad y conservación de la información en las empresas; Oracle brinda opciones para la realización e backup físicos y lógicos de la base de datos como RECOVERY MANAGER (RMAN) y ORACLE DATA PUMP EXPORT/IMPORT, así como mecanismos para configurar la base de datos de tal forma que sea más óptima su restauración con ARCHIVELOG.

4.1. CONFIGURACIÓN DE MODO ARCHIVELOG EN LA BASE DE DATOS.

Oracle cuenta con un mecanismo para respaldarse ante averías físicas del disco donde se guarda la base de datos, así como de modificaciones de los datos no autorizadas o no deseadas, ya sea por error de los usuarios, el administrador de la base de datos, o por debilidades en los aplicativos que interactúan con la base de datos o que no cuentan con suficientes controles para evitar que se realicen daños a los datos de forma involuntaria. Este mecanismo de respaldo se activa en la base de datos cuando es configurada en modo ARCHIVELOG, o modo de hacer copia a los archivos de redo log.

Según la documentación del proyecto AjpdSoft.Oracle dedicado a la publicación en internet de conocimientos en nuevas tecnologías, entre ellas base de datos, se describen ventajas que trae la configuración ARCHIVELOG en una base de datos Oracle, como la reducción de la posibilidad de pérdida de datos en caso de fallos, ya que hace posible que se puedan llevar a cabo restauraciones a la base de datos desde un punto específico de tiempo y permite realizar backup sin detener la base de datos (backup en caliente).

La documentación oficial de Oracle[92] hace referencia a que defecto la base de datos se instalan en modo NOARCHIVELOG. Entre las desventajas que trae el contar con una base de datos en modo NOARCHIVELOG, está que en caso de fallo que requiera restaurar la base de datos, se perderían las transacciones que se hayan realizado desde el último backup hasta el momento de la restauración. Otra desventaja de este modo de configuración es que sólo es posible hacer copias de seguridad con la base de datos cerrada.

La configuración de la base de datos en modo ARCHIVELOG, es recomendable para base de datos que operan 24 horas 7 días a la semana y para cuando se manejan datos muy críticos donde la menor perdida de información puede

[92] (ORACLE CORPORATION. Oracle Help Cente, Database Administrator's Guide: Choosing Between NOARCHIVELOG and ARCHIVELOG Mode. [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28310/archredo002.htm)

ocasionar inconvenientes graves a la compañía, según lo detalla la documentación oficial de Oracle[93].

Según la documentación del proyecto de publicación en internet de conocimientos en nuevas tecnologías AjpdSoft.Oracle, específicamente en base de datos Oracle, detalla los siguientes pasos para configurar la base de datos en archivelog[94]:

- En una consola de Sql Plus, conectarse a la Base de datos que se va a validar: *connect usuario/contraseña@[Nombre_BD] as sysdba*
- Inicialmente, se debe conocer si la base de datos está en modo Archivelog. Existen varias formas de hacer esta validación, como las siguientes:

  Se ingresa la siguiente instrucción en una consola en SQL Plus:

  SQL> archive log list;

  Un ejemplo de lo que arrojaría la consulta si la base de datos estuviera en modo NOARCHIVELOG:

  Database log mode No Archive Mode
  Automatic archival Disabled
  Archive destination /oracle10/product/10.1.3/dbs/arch
  Oldest online log sequence 36
  Current log sequence 38

  Otra forma de validar el modo de configuración, es consultando en la vista de parámetros de Oracle v$database, ejecutando la sentencia sql:

  select name, log_mode from v$database;

  Que arrojaría los posibles valores de *NOARCHIVELOG* o *ARCHIVELOG*

  También es posible validar el parámetro directamente en el archivo de configuración init.ora de la base de datos, donde el parámetro log_archive_start, debe estar en true

  log_archive_start = true

- Bajar la base de datos: shutdown immediate
- Montar la base de datos : startup mount
- Ejecutar la sentencia: alter database archivelog

[93] Ibíd.
[94] (PROYECTO AjpdSoft, : Activar modo ARCHIVELOG en Oracle Database 11g R2 Bases de Datos. [En línea]. 2016, Disponible en: http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=559)

- Abrir la base de datos: alter database open
- Activar el archivado automático: alter system archive log start
- shutdown immediate;

Para validar se ejecuta nuevamente la sentencia: Archive log list ó *select log_mode from v$database;*

4.2. RESPALDOS Y RESTAURACIÓN FÍSICOS CON RECOVERY MANAGER (RMAN)

Oracle permite la realización de respaldo y restauración de la base de datos a nivel físico mediante el uso de la Herramienta Recovery Manager (RMAN). Según la documentación oficial de Oracle[95], entre las principales características de esta herramienta se encuentran:

- Permite programar la automatización de tareas para la generación de backup y recuperación, creando archivos de texto que contenga los comandos rman necesarios para ejecutar estos procesos, a su vez que se configuran con opciones del sistema operativo. De esta forma se convierte en una aliada importante para la ejecución de las estrategias de respaldo que se planteen en la organización.
- Permite configurar backups incrementales, permitiendo realizar copias solamente de los bloques que hayan cambiado respecto al último backup realizado. (sólo en modo ARCHIVELOG). este tipo de configuración permite realizar copias y restauración de forma más rápida, ocupando menos espacio en disco.
- Es posible ejecutarlo por medio de línea de comandos rman o por entorno gráfico con la herramienta Oracle Enterprise Manager que facilita su administración.
- Permite detectar bloques corruptos desde el proceso de respaldo. obteniendo información de las vistas V$BACKUP_CORRUPTION y V$COPY_CORRUPTION o V$DATABASE_BLOCK_CORRUPTION.
- Permite la realización de Backus en frio o en caliente , este último solo cuando la base de datos está configurada en modo ARCHIVELOG

[95] (ORACLE CORPORATION.Oracle Help Centrer, Database Backup and Recovery User's Guide: 1 Introduction to Backup and Recovery [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRADV8001)

Según la documentación oficial de Oracle[96], los comandos RMAN más comunes para ejecutar tareas de respaldo y restauración:

- Para comenzar a ejecutar comandos rman, desde una ventana de línea de comandos del sistema operativo se ejecuta 'rman', de esta forma el prompt cambia a RMAN>, y ya es posible enviar comandos propios de rman para ejecutar los diferentes procesos de respaldo y restauración de la base de datos.

Para conectarse a la base de datos desde rman:

- Con el usuario SYS de la base de datos:

RMAN> CONNECT TARGET SYS/pwd@prod

- Con el usuario del sistema operativo:

RMAN> CONNECT TARGET /

Entre los comandos RMAN más comunes a ejecutar están:

- Para conocer la configuración de rman en la base de datos:

RMAN> SHOW ALL;

- Copia de base de datos y archivos regdo logs, estando configurada la base de datos en modo ARCHIVELOG:

RMAN> BACKUP DATABASE PLUS ARCHIVELOG;

- Realizar un backup de la base de datos:

RMAN> BACKUP DATABASE;

- Realizar imagen del backup de todos los archivos de la base de datos.

RMAN> BACKUP AS COPY DATABASE;

- Para validar posibles backups corruptos incluyendo los redo log files.

[96] (ORACLE CORPORATION.Oracle Help Centrer, Database Backup and Recovery User's Guide: 9 Backing Up the Database [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmbckba.htm#BRADV8003)

BACKUP VALIDATE
DATABASE
ARCHIVELOG ALL;

- Visualizar bloques corruptos en las copias

SQL> SELECT * FROM V$DATABASE_BLOCK_CORRUPTION;

- Para listar las copias de la base de datos realizadas

RMAN> LIST BACKUP;
RMAN> LIST COPY;
RMAN>LIST BACKUP SUMMARY;

- Para salir de rman

RMAN> EXIT

- Un ejemplo de un archivo creado con alguna tarea de backup y su ejecución es el siguiente:

En un editor de texto se escribe:
 # archivo bkrman.txt
CONNECT TARGET /
BACKUP DATABASE PLUS ARCHIVELOG;
LIST BACKUP;
EXIT;

Para ejecutarlo, desde línea de comandos en el sistema operativo:
% rman @/my_dir/bkrman.txt

4.3. RESPALDOS Y RESTAURACIÓN LÓGICOS CON ORACLE DATA PUMP EXPORT/IMPORT

Para realizar respaldos y restauración lógica de la base de datos, según la documentación oficial de Oracle[97], Oracle 11g cuenta con la funcionalidad de export/import datapump. Esta herramienta permite realizar backup lógicos ya sea de toda la base de datos, esquemas, u objetos específicos como tablas, procedimientos, paquetes, de una forma sencilla, creando unos archivos de fácil portabilidad. Sólo funciona con la base de datos abierta.

Para realizar respaldo con export datapump:

- ✓ Como Export FULL, se ejecuta la instrucción:
  expdp userid=$DBUSER/$DBPSWD dumpfile=$FILE.dmp logfile=$FILE.log full=yes
- ✓ Como export de un esquema, se ejecuta la instrucción:
  expdp userid=$DBUSER/$DBPSWD dumpfile=$FILE.dmp logfile=$FILE.log schemas=$SCHEMA

4.4 CONFIGURACIÓN DE POLÍTICAS DE RESPALDO

Dependiendo de tamaño de BD y criticidad, se deben contar con planes de copias de seguridad y recuperación, que estén acorde a la continuidad que se espera del negocio. Un ejemplo de política podría ser:

- Respaldos con export datapump:
  - De Lunes a Viernes a las 12:30 PM
  - De Lunes a Viernes a las 6:45 PM
- Respaldos con RMAN:
  - Todos los días cada hora, backup de archive logs
  - Todos los días a las 11:30 PM, full backup

[97] (ORACLE CORPORATION.Oracle Help Centrer, Database Utilities: 2 Data Pump Export [En línea]. 2016, Disponible en: https://docs.oracle.com/cd/B28359_01/server.111/b28319/dp_export.htm)

4.5. POLÍTICAS DE SEGURIDAD RELACIONADAS CON RESPALDO Y RESTAURACIÓN

- ✓ Dentro de la segregación de funciones en la Base de datos, se debe contar con un usuario que solo tenga privilegios para la generación de copias de seguridad y restauración. Esto debido a que las funciones de administración de la base de datos en una organización, en ocasiones se cuenta con varios responsables con diferentes funciones: monitoreo, gestión de usuarios, y encargado de las copias de seguridad.
- ✓ Incluir en el plan de recuperación los pasos a seguir en caso de una incidencia critica en la BD.
- ✓ Se debe contar con tareas de validación de los BK generados y cintas grabadas, para medir la calidad de los ficheros BK, y tiempos de respuesta ante un suceso de alerta.
- ✓ Las cintas debe almacenarse en un lugar diferente a las instalaciones donde se encuentre el servidor de la base de datos.

ANEXO C – AUTORIZACIÓN DE EJECUCIÓN DE PRUEBAS EN LA BASE DE DATOS DE DESARROLLO ORACLE 11G DE LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC)

Corporación Autónoma
Regional del Valle del Cauca

Cali, 8 de septiembre de 2016

Señores
Universidad Nacional Abierta y a Distancia
UNAD.

La jefatura de la oficina de tecnologías de la información de la Corporación Autónoma Regional del Valle del Cauca (CVC), autoriza a Gustavo Adolfo Herrera Angola identificado con cedula No. 94330824 y Margarita leal Joya identificada con cedula No. 63450413, a realizar pruebas de configuración de seguridad de la base de datos de la Corporación como parte de las actividades incluidas en el desarrollo del proyecto de grado titulado "DISEÑO DE UNA GUÍA PARA LA ADMINISTRACIÓN DE LA SEGURIDAD DE BASES DE DATOS EN UN ENTORNO DE ORACLE 11G, APLICADA A LA CORPORACIÓN AUTÓNOMA REGIONAL DEL VALLE DEL CAUCA (CVC) EN LA CIUDAD DE CALI", para optar al título de Especialistas en Seguridad Informática de la Universidad Nacional Abierta y a Distancia UNAD.

La práctica se autoriza realizar bajo las siguientes condiciones:

- Se autoriza realizarla en la base de datos de desarrollo, la cual se encuentra configurada como una base de datos espejo de la base de datos de producción.
- No se podrá llevar a cabo ningún tipo de práctica directamente en ambientes de producción. De esta forma prevenir riesgos de alteraciones al normal funcionamiento tecnológico de la CVC, que puedan acarrear inconvenientes a la oficina de tecnología responsable en la corporación del funcionamiento y soporte de toda la infraestructura tecnológica de la empresa.
- Se requiere total confidencialidad de la información correspondiente a parámetros de la base de datos, nombre de host, IP, nombre de objetos, instancias. Quedando prohibido la divulgación de este tipo de información.
- Sólo se autoriza la divulgación parcial de los resultados arrojados en la valoración de seguridad que se lleve a cabo, prohibiendo divulgación de información explicita de la base de datos, que coloquen en riesgo o pueda ser utilizada por terceros para acciones malintencionadas.

Carrera 56 11-36
Santiago de Cali, Valle del Cauca
PBX: 620 66 00 – 3181700
Línea verde: 018000933093
atencionalusuario@cvc.gov.co
www.cvc.gov.co

- La información detallada de los resultados obtenidos, deben ser entregados y socializados con la jefatura de sistemas y administrador de la base de datos.

Diego Alexander Millan Londoño
Jefe Oficina de Tecnología de la Información

Carrera 56 11-36
Santiago de Cali, Valle del Cauca
PBX: 620 66 00 – 3181700
Línea verde: 018000933093
atencionalusuario@cvc.gov.co
www.cvc.gov.co